19th Conference on the Theory of Quantum Computation, Communication and Cryptography

TQC 2024, September 9–13, 2024, Okinawa, Japan

Edited by

Frédéric Magniez
Alex Bredariol Grilo

LIPICS

Editors

Frédéric Magniez
Université Paris Cité, CNRS, IRIF, Paris, France
frederic.magniez@irif.fr

Alex Bredariol Grilo
LIP6, Paris, France
Sorbonne Université, Paris, France
CNRS, Paris, France
alex.bredariol-grilo@lip6.fr

ACM Classification 2012
Theory of computation → Quantum computation theory; Theory of computation → Quantum complexity theory; Theory of computation → Quantum query complexity; Theory of computation → Quantum information theory; Theory of computation → Cryptographic protocols; Theory of computation → Cryptographic primitives; Hardware → Quantum computation; Computer systems organization → Quantum computing

ISBN 978-3-95977-328-7

Published online and open access by
Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at https://www.dagstuhl.de/dagpub/978-3-95977-328-7.

Publication date
September, 2024

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at https://portal.dnb.de.

Digital Object Identifier: 10.4230/LIPIcs.TQC.2024.0

ISBN 978-3-95977-328-7 **ISSN 1868-8969** **https://www.dagstuhl.de/lipics**

LIPIcs – Leibniz International Proceedings in Informatics

LIPIcs is a series of high-quality conference proceedings across all fields in informatics. LIPIcs volumes are published according to the principle of Open Access, i.e., they are available online and free of charge.

ISSN 1868-8969

https://www.dagstuhl.de/lipics

Contents

Preface
Frédéric Magniez and Alex Bredariol Grilo 0:vii

Conference Organization
. . . . 0:ix

Outstanding Paper Award
. . . . 0:xiii

List of Authors
. . . . 0:xv

Papers

Multi-qubit Lattice Surgery Scheduling
Allyson Silva, Xiangyi Zhang, Zak Webb, Mia Kramer, Chan-Woo Yang, Xiao Liu, Jessica Lemieux, Ka-Wai Chen, Artur Scherer, and Pooya Ronagh 1:1–1:22

Stochastic Error Cancellation in Analog Quantum Simulation
Yiyi Cai, Yu Tong, and John Preskill 2:1–2:15

Efficient Optimal Control of Open Quantum Systems
Wenhao He, Tongyang Li, Xiantao Li, Zecheng Li, Chunhao Wang, and Ke Wang 3:1–3:23

One-Wayness in Quantum Cryptography
Tomoyuki Morimae and Takashi Yamakawa 4:1–4:21

Revocable Quantum Digital Signatures
Tomoyuki Morimae, Alexander Poremba, and Takashi Yamakawa 5:1–5:24

The Quantum Decoding Problem
André Chailloux and Jean-Pierre Tillich 6:1–6:14

Eigenpath Traversal by Poisson-Distributed Phase Randomisation
Joseph Cunningham and Jérémie Roland 7:1–7:20

(Quantum) Complexity of Testing Signed Graph Clusterability
Kuo-Chin Chen, Simon Apers, and Min-Hsiu Hsieh 8:1–8:16

Quantum Non-Identical Mean Estimation: Efficient Algorithms and Fundamental Limits
Jiachen Hu, Tongyang Li, Xinzhao Wang, Yecheng Xue, Chenyi Zhang, and Han Zhong 9:1–9:21

Guidable Local Hamiltonian Problems with Implications to Heuristic Ansatz State Preparation and the Quantum PCP Conjecture
Jordi Weggemans, Marten Folkertsma, and Chris Cade 10:1–10:24

A Direct Reduction from the Polynomial to the Adversary Method
Aleksandrs Belovs 11:1–11:15

Quantum Delegation with an Off-The-Shelf Device
Anne Broadbent, Arthur Mehta, and Yuming Zhao 12:1–12:23

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Preface

The 19th Conference on The Theory of Quantum Computation, Communication and Cryptography (TQC) was hosted by the Okinawa Institute for Science and Technology in Japan, and held from September 9 to September 13, 2024.

The TQC conference is a leading annual international conference for students and researchers working in the theoretical aspects of quantum information science. The scientific objective of TQC is to bring together the theoretical quantum information science community to present and discuss the latest advances in the field.

Areas of interest for TQC include, but are not restricted to: quantum algorithms, models of quantum computation, quantum complexity theory, simulation of quantum systems, quantum cryptography, quantum communication, quantum information theory, quantum estimation and measurement, quantum error correction and fault-tolerant quantum computing, intersection of quantum information and condensed-matter theory, intersection of quantum information and machine learning.

A list of the previous editions of TQC follows:

- TQC 2023, University of Aveiro, Portugal
- TQC 2022, University of Illinois at Urbana-Champaign, USA
- TQC 2021, University of Latvia, Latvia (virtual conference)
- TQC 2020, University of Latvia, Latvia (virtual conference)
- TQC 2019, University of Maryland, USA
- TQC 2018, University of Technology Sydney, Australia
- TQC 2017, Université Pierre et Marie Curie, France
- TQC 2016, Freie Universität Berlin, Germany
- TQC 2015, Université libre de Bruxelles, Brussels, Belgium
- TQC 2014, National University of Singapore, Singapore
- TQC 2013, University of Guelph, Canada
- TQC 2012, University of Tokyo, Japan
- TQC 2011, Universidad Complutense de Madrid, Spain
- TQC 2010, University of Leeds, UK
- TQC 2009, Institute for Quantum Computing, University of Waterloo, Canada
- TQC 2008, University of Tokyo, Japan
- TQC 2007, Nara Institute of Science and Technology, Nara, Japan
- TQC 2006, NTT R&D Center, Atsugi, Kanagawa, Japan

The conference consisted of invited talks, contributed talks, a poster session, a rump session, and a business meeting. The invited talks were given by Jens Eisert (FU Berlin), Zhengfeng Ji (Tsinghua University), Dakshita Khurana (University of Illinois Urbana-Champaign), and Tomoyuki Morimae (Yukawa Institute for Theoretical Physics, Kyoto University). Submissions were solicited for two tracks: *With Proceedings* (talk and proceedings) and *Without Proceedings* (talk only).

There were 460 submissions for talks, 44 of which were also submitted to the *With Proceedings* track. The program committee selected 92 submissions for talks, including 12 from the *With Proceedings* track. This year, the program committee also selected 19 submissions for outstanding posters.

We wish to thank the members of the Program Committee and all subreviewers for their incredible work towards composing the program of the conference. We would also like to thank the Local Organizing Committee for all their efforts in organizing the conference, as well as the Steering Committee for maintaining the conference's high standards. Last but not least, we thank the authors of all the TQC 2024 submissions.

Conference Organization

Organizing Committee

Local organizers in Okinawa

- David Elkouss Coronas, OIST
- Kae Nemoto, OIST
- Slawomir Rosiek, OIST
- Yukari Yoseda, OIST

International organizers

- Lídia del Rio, Squids and University of Zurich
- Nuriya Nurgalieva, Squids and University of Zurich

Program Committee

- Srinivasan Arunachalam, IBM
- Alexander Belovs, University of Latvia
- Mario Berta, RWTH Aachen University
- Xavier Bonnetain, Inria Nancy
- Jop Briet, CWI
- Alex Bredariol Grilo, CNRS [co-chair]
- Marco Cerezo, LANL
- Nai-Hui Chia, Rice University
- Nicolas Delfosse, IonQ
- Ernesto Galvão, INL
- Uma Girish, Princeton
- Tom Gur, University of Cambridge
- Yassine Hamoudi, CNRS Bordeaux
- Dominik Hangleiter, QuICS (UMD & NIST)
- Chris Heunen, University of Edinburgh
- Christoph Hirche, TU Munich and CQT NUS
- Nick Hunter-Jones, UT Austin
- John Kallaugher, Sandia National Laboratories
- Shelby Kimmel, Middlebury College
- Robert Koenig, TU Munich
- Felix Leditzky, UIUC
- Tongyang Li, Peking University
- Jiahui Liu, MIT
- Frédéric Mangiez, CNRS [chair]
- Alex May, Perimeter Institute and University of Waterloo
- Mio Murao, University of Tokyo
- Ion Nechita, CNRS, Toulouse
- Harumichi Nishimura, Nagoya University
- Tom O'Brien, Google Quantum AI
- Subhasree Patro, Utrecht University and QuSoft

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- Supartha Podder, Stony Brook University
- Alexander Poremba, MIT
- Luowen Qian, Boston University
- Patrick Rebentrost, CQT
- Norbert Schuch, University of Vienna
- Thomas Schuster, Caltech
- Makrand Sinha, UIUC
- Fang Song, Portland State University
- David Sutter, IBM Zurich
- Mario Szegedy, Rutgers University
- Marcelo Terra Cunha, Unicamp
- Dave Touchette, Sherbrooke University
- Dominic Verdon, University of Bristol
- Nathan Wiebe, University of Toronto
- Dominic Williamson, University of Sydney
- Penghui Yao, Nanjing University
- Ted Yoder, IBM

Steering Committee

- Andris Ambainis, University of Latvia
- Eric Chitambar, University of Illinois at Urbana-Champaign
- Kai-Min Chung, Academia Sinica
- Steve Flammia, AWS Center for Quantum Computing
- François Le Gall, Nagoya University [co-chair]
- Min-Hsiu Hsieh, Hon Hai (Foxconn) [chair]
- Kae Nemoto, OIST
- Lídia del Rio, Squids and University of Zurich

Subreviewers

H. Aaronson
A. Abbas
Y. Ai
A. Alhambra
R. Allerstorfer
Y. Alnawakhtha
D. An
M. Anastasia Jivulescu
A. Angrisani
E. Anschuetz
G. Anselmetti
A. Anshu
M. Arienzo
A. Arqand
F. Baccari
A. Baczewski
H. Badhani
Z. Baghali Khanian
C. Bai
S. Balasubramanian
L. Banchi
J. Bao
Y. Bao
Z. Bao
J. Bavaresco
A. Bellante
A. Bene-Watts
A. Benhemou
D. Bera
L. Berent
T. Bergamaschi
B. Bergh
P. Bermejo
M. Beverland
K. Bharti
M. Black
V. Blakaj
A. Block
M. Block
A. Bluhm
N. Boddu
X. Bonet-Monroig
J. Bostanci
P. Botteron
P. Braccia
C. Branciard
L. Brenner
A. Broadbent
D. Brod
B. Brown
P. Brown
D. Browne
K. Bu
J. Bulmer
A. Burchardt
L. Burri
Z. Cai
A. Çakan
M. Caro
J. Carolan
L. Catani
E. Cervero
U. Chabaud
S. Chakraborty
R. Chatterjee
A. Chaturvedi
B. Chen
C. Chen
S. Chen
Y. Chen
B. Cheng
S. Chessa
C. Chubb
L. Cincio
J. Claes
N. Coble
L. Cohen
A. Coladangelo
L. Colisson
J. Conrad
A. Cornelissen
J. Crann
E. Culf
P. Czarnik
A. Dalzell
A. Darmawan
I. Datta
G. Dauphinais
M. Davydova
G. De Palma
J. de Vicente
C. Derby
E. Derbyshire
A. Deshpande
S. Designolle
B. Dias
N. Diaz
B. Doolittle
M. Doosti
J. Doriguello
R. Drumond
M. Duschenes
A. Dutkiewicz
A. Dutt
D. Egger
T. Ellison
E. Epperly
F. Escudero-Gutiérrez
I. Faisal
D. Fang
M. Fanizza
O. Fawzi
B. Fefferman
S. Oliviero
H. Fu
M. Gachechiladze
M. Gao
D. García-Martín
R. Garcia-Patron Sanchez
J. Garre
G. Gentinetta
I. George
M. Gessner
S. Ghosh
L. Giannelli
A. Gilani
V. Gitton
M. Goh
L. Golowich
P. Gondolf
C. Gonzalez-Guillen
K. Goodenough
A. Gopal Maity
N. Goud Boddu
A. Green
S. Grewal
D. Grier
D. Grinko
J. Gross
V. Guemard
A. Gulati
N. Guo
A. Gupte
C. Gyurik
E. Haapasalo
Z. Han
A. Hasegawa
V. Havlicek
M. Hayashi
Z. He
M. Heinrich
J. Helsen
M. Hhan
T. Hillmann
M. Hinsche
S. Ho Choe
M. Hoban
B. Holman
Z. Holmes
D. Hothem
Y. Hu
H. Huang
P. Huang
Q. Huang
Y. Huang
F. Huber
T. Huffstutler
F. Hufnagel
W. Huggins
S. Hung
Y. Hwang
J. Iosue
M. Ippoliti
J. Iverson
P. Iyer
V. Iyer
W. J. Huggins
M. Jabbour
A. Jain
S. Jain
F. Jeronimo
A. Jha
J. Jiang
Z. Jiang
T. Jochym-O'Connor
P. Johnson
N. Ju
H. Kadri
G. Kahanamoku-Meyer
F. Kaleoglu
L. Kamin
J. Kamminga
M. Kang
U. Kapshikar
S. Kar
K. Kato
A. Kawachi
Z. Khanian
R. King
F. Kitagawa
B. Kobrin
T. Kohler
N. Kornerup
G. Koßmann
L. Kovalsky
W. Kretschmer
H. Krovi
A. Kubica
R. Kunjwal
T. Kuwahara
C. Lai
C. Lancien
N. Laracuente
M. Larocca
D. Layden
Y. Le Borgne
E. Lee
M. Lehmkühler
J. Leng
L. Leppäjärvi
A. Leverrier
K. Li
X. Li
Z. Li
D. Liang
X. Liang
D. Lim
C. Lin
H. Lin
J. Lin
Y. Lin
D. Litinski
J. Liu
Q. Liu
Y. Liu
Z. Liu
S. Llorens
F. Loulidi
Y. Lu
M. Luce
J. Lumbreras
F. Maciejewski
J. Magdalena
S. Majidy
N. Mande
A. Mantri
M. Marvian
K. Marwaha
N. Maskara
J. McClean
C. McLauchlan
S. Mehraban
R. Meister
C. Mendl
T. Metger
J. Meyer
D. Miloschewsky
K. Miyamoto
T. Möbus
A. Molnar
J. Mora
G. Morais
M. Morales
T. Morimae
H. Mousavi
G. Muguruza Lasa
S. Mutreja
G. Nannicini
V. Narasimhachar
H. Nator
M. Navascués
A. Nayak
B. Nehoran
A. Nema
R. Nery
N. Neumann
Q. Nguyen
A. Nietner
C. Nirkhe
P. Niroula
E. Onorati
D. Orsucci
A. Oufkir
Y. Ouyang
M. Ozols
C. Paddock
F. Pan
O. Parekh
N. Parham
S. Park
G. Pass
C. Pattison
A. Pelecanos
A. Pérez-Salinas
A. Pesah
C. Piveteau
S. Polla
C. Porto
A. Pozas-Kerstjens
N. Pranzini
S. Puri
Y. Quek
A. Quintavalle
S. Ragavan
H. Rall
P. Rall
C. Ramanthanan
S. Rao
C. Rayudu
B. Regula
N. Rengaswamy
N. Resch
V. Reza Asadi
T. Rippchen
B. Roberts
S. Roberts
D. Rochette
G. Rosenthal
Z. Rossi
I. Roth
C. Rouzé
B. Royer
R. Rubboli
N. Rubin
K. Rudinger
M. Rudolph
D. Ruiz
D. Saha
S. Sajjad Nezhadi
F. Salek
W. Salmon
R. Salzmann
M. Sandfuchs
R. Santagati
R. Sarkar
M. Sarovar
O. Sattath
F. Sauvage
L. Schaeffer
L. Schatzki
E. Schoute
A. Schrottenloher
A. Seif
R. Sengupta
B. Senjean
K. Senthoor
C. Shao
S. Shao
A. She
Y. Shen
K. Shi
O. Shtanko
V. Siddhu
D. Silva
J. Slote
R. Soares Barbosa
M. Soleimanifar
R. Spekkens
S. Sreekumar
T. Steckmann
D. Stephen
D. Stilck França
M. Streif
A. Streltsov
A. Strikis
M. Studzinski
Y. Su
P. Suchsland
L. Sun
I. Supic
R. Sweke
K. Szymanski
F. Tacchino
M. Tahmasbi
R. Takagi
X. Tan
E. Tang
T. Temistocles
M. Terra Cunha
S. Thanasilp
A. Tikku
I. Todinca
M. Tomamichel
K. Tomer
Y. Tong
M. Túlio Quintino
Q. Tupker
J. Tura Brugués
C. Tüysüz
P. U Rao
D. Underwood
V. V. Albert
E. van der Berg
V. Vandaele
B. Varbanov
F. Vasconcelos
M. Vasmer
M. Vempati
B. Vermersch
C. Vieira
A. Villanyi
N. Voronova
C. Vuillot
R. Wagner
C. Wang
D. Wang
J. Wang
Q. Wang
S. Wang
T. Wang
X. Wang
Y. Wang
M. Weilenmann
R. Wiersema
D. Wild
J. Wilkens
A. Wills
P. Wocjan
K. Wu
P. Wu
X. Wu
Y. Wu
Q. Xu
Y. Xue
E. Y.-Z.Tan
A. Yakaryilmaz
T. Yamakawa
H. Yamasaki
R. Yang
Y. Yang
Y. Yao
B. Ye
B. Yee Gan
J. Yirka
N. Yoshioka
Z. Yu
X. Yuan

Outstanding Paper Award

From the submission of the track *With Proceedings*, the Program Committee selected as the TQC 2024 Outstanding Papers, by order of publication in the proceedings:

- *Multi-qubit Lattice Surgery Scheduling*, by Allyson Silva, Xiangyi Zhang, Zachary Webb, Mia Kramer, Chan Woo Yang, Xiao Liu, Jessica Lemieux, Kawai Chen, Artur Scherer, Pooya Ronagh
- *Stochastic error cancellation in analog quantum simulation*, by Yiyi Cai, Yu Tong, John Preskill

List of Authors

Simon Apers (8)
Université de Paris, CNRS, IRIF, France

Aleksandrs Belovs (11)
Faculty of Computing, University of Latvia,
Riga, Latvia

Anne Broadbent (12)
Department of Mathematics and Statistics,
University of Ottawa, Canada;
Nexus for Quantum Technologies,
University of Ottawa, Canada

Chris Cade (10)
Fermioniq, Amsterdam, The Netherlands;
QuSoft & University of Amsterdam (UvA),
The Netherlands

Yiyi Cai (2)
Institute for Quantum Information and Matter,
California Institute of Technology, Pasadena,
CA, USA;
Department of Electrical Engineering, California
Institute of Technology, Pasadena, CA, USA

André Chailloux (6)
Inria de Paris, France

Ka-Wai Chen (1)
1QB Information Technologies (1QBit),
Vancouver, Canada

Kuo-Chin Chen (8)
Hon Hai Research Institute, Taipei, Taiwan

Joseph Cunningham (7)
Centre for Quantum Information and
Communication (QuIC), Ecole polytechnique de
Bruxelles, Université libre de Bruxelles, Belgium

Marten Folkertsma (10)
CWI & QuSoft, Amsterdam, The Netherlands

Wenhao He (3)
Center for Computational Science and
Engineering, MIT, Cambridge, MA, USA;
School of Physics, Peking University, Beijing,
China

Min-Hsiu Hsieh (8)
Hon Hai Research Institute, Taipei, Taiwan

Jiachen Hu (9)
Peking University, Beijing, China

Mia Kramer (1)
1QB Information Technologies (1QBit),
Vancouver, Canada

Jessica Lemieux (1)
1QB Information Technologies (1QBit),
Vancouver, Canada

Tongyang Li (3, 9)
Center on Frontiers of Computing Studies,
School of Computer Science, Peking University,
Beijing, China

Xiantao Li (3)
Department of Mathematics, Pennsylvania State
University, University Park, PA, USA

Zecheng Li (3)
Department of Computer Science and
Engineering, Pennsylvania State University,
University Park, PA, USA

Xiao Liu (1)
1QB Information Technologies (1QBit),
Vancouver, Canada

Arthur Mehta (12)
Department of Mathematics and Statistics,
University of Ottawa, Canada; Nexus for
Quantum Technologies, University of Ottawa,
Canada

Tomoyuki Morimae (4, 5)
Yukawa Institute for Theoretical Physics,
Kyoto University, Japan

Alexander Poremba (5)
Computing and Mathematical Sciences, Caltech,
Pasadena, CA, USA;
CSAIL and Department of Mathematics, MIT,
Cambridge, MA, USA

John Preskill (2)
Institute for Quantum Information and Matter,
California Institute of Technology, Pasadena,
CA, USA;
AWS Center for Quantum Computing,
Pasadena, CA, USA

Jérémie Roland (7)
Centre for Quantum Information and
Communication (QuIC), Ecole polytechnique de
Bruxelles, Université libre de Bruxelles, Belgium

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Pooya Ronagh (1)
1QB Information Technologies (1QBit), Vancouver, Canada;
Institute for Quantum Computing, University of Waterloo, Canada;
Department of Physics & Astronomy, University of Waterloo, Canada;
Perimeter Institute for Theoretical Physics, Waterloo, Canada

Artur Scherer (1)
1QB Information Technologies (1QBit), Vancouver, Canada

Allyson Silva (1)
1QB Information Technologies (1QBit), Vancouver, Canada

Jean-Pierre Tillich (6)
Inria de Paris, France

Yu Tong (2)
Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA, USA

Chunhao Wang (3)
Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA

Ke Wang (3)
Department of Mathematics, Pennsylvania State University, University Park, PA, USA

Xinzhao Wang (9)
Peking University, Beijing, China

Zak Webb (1)
1QB Information Technologies (1QBit), Vancouver, Canada

Jordi Weggemans (10)
CWI & QuSoft, Amsterdam, The Netherlands;
Fermioniq, Amsterdam, The Netherlands

Yecheng Xue (9)
Peking University, Beijing, China

Takashi Yamakawa (4, 5)
NTT Social Informatics Laboratories, Tokyo, Japan;
NTT Research Center for Theoretical Quantum Information, Atsugi, Japan;
Yukawa Institute for Theoretical Physics, Kyoto University, Japan

Chan-Woo Yang (1)
1QB Information Technologies (1QBit), Vancouver, Canada

Chenyi Zhang (9)
Stanford University, CA, USA

Xiangyi Zhang (1)
1QB Information Technologies (1QBit), Vancouver, Canada

Yuming Zhao (12)
Institute for Quantum Computing, University of Waterloo, Canada;
Department of Pure Mathematics, University of Waterloo, Canada

Han Zhong (9)
Peking University, Beijing, China

Multi-qubit Lattice Surgery Scheduling

Allyson Silva ✉
1QB Information Technologies (1QBit), Vancouver, Canada

Xiangyi Zhang ✉
1QB Information Technologies (1QBit), Vancouver, Canada

Zak Webb ✉
1QB Information Technologies (1QBit), Vancouver, Canada

Mia Kramer ✉
1QB Information Technologies (1QBit), Vancouver, Canada

Chan-Woo Yang ✉
1QB Information Technologies (1QBit), Vancouver, Canada

Xiao Liu
1QB Information Technologies (1QBit), Vancouver, Canada

Jessica Lemieux
1QB Information Technologies (1QBit), Vancouver, Canada

Ka-Wai Chen
1QB Information Technologies (1QBit), Vancouver, Canada

Artur Scherer ✉
1QB Information Technologies (1QBit), Vancouver, Canada

Pooya Ronagh[a] ✉
1QB Information Technologies (1QBit), Vancouver, Canada
Institute for Quantum Computing, University of Waterloo, Canada
Department of Physics & Astronomy, University of Waterloo, Canada
Perimeter Institute for Theoretical Physics, Waterloo, Canada

[a] Corresponding author

Abstract

Fault-tolerant quantum computation using two-dimensional topological quantum error correcting codes can benefit from multi-qubit long-range operations. By using simple commutation rules, a quantum circuit can be transpiled into a sequence of solely non-Clifford multi-qubit gates. Prior work on fault-tolerant compilation avoids optimal scheduling of such gates since they reduce the parallelizability of the circuit. We observe that the reduced parallelization potential is outweighed by the significant reduction in the number of gates. We therefore devise a method for scheduling multi-qubit lattice surgery using an earliest-available-first policy, solving the associated forest packing problem using a representation of the multi-qubit gates as Steiner trees. Our extensive testing on random and various Hamiltonian simulation circuits demonstrates the method's scalability and performance. We show that the transpilation significantly reduces the circuit length on the set of circuits tested, and that the resulting circuit of multi-qubit gates has a further reduction in the expected circuit execution time compared to serial execution.

2012 ACM Subject Classification Computer systems organization → Quantum computing; Software and its engineering → Compilers; Hardware → Quantum error correction and fault tolerance; Software and its engineering → Scheduling; Hardware → Circuit optimization

Keywords and phrases Scheduling, Large-Scale Optimization, Surface Code, Quantum Compilation, Circuit Optimization

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.1

Related Version *Full Version*: `https://arxiv.org/abs/2405.17688`

Supplementary Material *Dataset (Quantum circuits used for the experiments)*: `https://doi.org/10.5281/zenodo.11391890`

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 1; pp. 1:1–1:22
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Funding *Pooya Ronagh*: Acknowledges the financial support of Mike and Ophelia Lazaridis, Innovation, Science and Economic Development Canada (ISED), and the Perimeter Institute for Theoretical Physics. Research at the Perimeter Institute is supported in part by the Government of Canada through ISED and by the Province of Ontario through the Ministry of Colleges and Universities.

Acknowledgements We are grateful to our editor, Marko Bucyk, for his careful review and editing of the manuscript.

1 Introduction

Fault-tolerant quantum computation (FTQC) aims to ensure reliable quantum computing despite faulty physical qubits. In FTQC, quantum error correction (QEC) is used to protect a logical Hilbert space within a much larger one. Topogical quantum error correcting codes in two dimensions, such as surface codes [8], are of particular interest given the convenience of nearest neighbour interactions for physical realization of quantum computers. FTQC can be achieved on topological error correction codes using *lattice surgery*, which facilitates long-range entanglement via auxiliary topological patches [12].

At the physical level, the circuits executed during FTQC, are repeated rounds of parity check operations that are scheduled to perform desired logical gates. The logical qubits (codes) that are not involved in a logical gate must still be protected using rounds of parity checks. Therefore, minimizing the depth of the logical circuit by parallelizing the gates reduces the total accumulated error during computation. Our work addresses the problem of scheduling these quantum operations on a fault-tolerant architecture using lattice surgery, which we refer to as the *lattice surgery scheduling problem* (LSSP). Efficient methods for solving the LSSP not only serve as foundations for future quantum compilers but are also immediately applicable for predicting the quantum resources required for target quantum algorithms. We will focus on surface codes in the rest of this paper; however, our approach to the LSSP is easily generalizable to other two-dimensional topological codes.

A fault-tolerant algorithm can be represented as a sequence of Clifford and non-Clifford Pauli rotations [17]. The Clifford gates are commuted to the end of the circuit and past the logical measurements, resulting in a solely non-Clifford sequence of logical gates. We call this step transpilation. This procedure is perceived to have two drawbacks: (1) it is computationally expensive to iteratively apply a set of commutation rules to pairwise consecutive gates to achieve this circuit, and (2) the resulting non-Clifford gates are highly non-local and therefore less parallelizable. The latter caveat motivates [2, 26] to avoid this transpilation and use algorithms for solving various shortest path problems to parallelize circuits involving single- and two-qubit gates.

The first drawback can be rectified using a result known to the community (explicitly explained in Appendix D of [13]) using the symplectic representation of Clifford gates to implement an efficient transpiler. By sweeping over the entire circuit, a symplectic representation is updated through commutation events. This procedure scales linearly with the total number of logical gates (as opposed to quadratic scaling of naïve usage of pairwise commutation rules). Then commuting layers of non-Clifford gates are formed, and used to combine some of these gates into Clifford ones. The procedure of commuting Clifford operations out is then repeated on the new sequence, until convergence is achieved.

As for the second perceived drawback, we show in Section 5 that the reduction in gate count from the transpilation process greatly exceeds the reduction in parallelizability of realistic circuits. We also observe that there is still a significant parallelizability potential between the highly non-local resultant gates, motivating us to solve the LSSP by devising greedy heuristics for solving the NP-hard terminal Steiner tree problem [16]. We do so by decomposing the LSSP into forest packing problems, another variant in the Steiner tree problem family [15, 9]. This entails generating Steiner trees that connect the qubits required by each parallelized operation without involving overlapping resources.

Our scheduling algorithm's performance is evaluated on a diverse set of circuits, including some generated to simulate real quantum systems, such as in the field of quantum chemistry, with up to nearly 23 million gates prior to optimization. We analyze the scalability and performance of our proposed algorithms to reduce gate count and to schedule operations. Our algorithm also allows us to compare the performance of various layouts of arrays of logical qubits surrounded by bus qubits (see Section 2 for more details). As a corollary, we propose a layout that provides a good balance for the space–time cost trade-off against previously suggested layouts. We draw the following conclusions from our study:

- The transpilation algorithm [17] for gate count reduction will be essential for enabling large-scale FTQC as it reduces circuit length by around one order of magnitude for the circuits tested.
- Despite the transpilation reducing parallelizability of operations, the resulting circuits do not require prohibitive runtimes, unlike as stated in [2]. Across all tested circuits, lower bounds calculated for the optimal number of logical cycles required to run pre-transpiled circuits are between about two and 12 times higher than upper bounds found for post-transpiled ones.
- Our proposed algorithm can schedule the multi-qubit gates at a rate of tens of thousands of operations per second in the computational environment tested, meaning that large quantum circuits can be scheduled in between a few minutes and a couple of hours using the proposed method.
- The parallel scheduling also results in solutions that are better than those of serial scheduling, with some circuits among those tested having as many as a third of their operations benefiting from parallelization, while others have as few as 0.1%.

This paper is organized as follows. Section 2 presents the surface code layouts studied, which are necessary for understanding the scheduling problem we solve. In Section 3, we mathematically define the LSSP using a decomposition method which guided us in designing our heuristics. Section 4 describes the algorithms proposed to generate dependency constraints and to solve the LSSP. In Section 5, we present the results and an analysis of our computational experiments and assess the performance of the proposed algorithms for a variety of circuits. We conclude the paper in Section 6 with some remarks on our research.

2 The surface code layout

Following [17], a circuit described in the Clifford + T gate set consisting of the Pauli gates (X, Y, Z), Hadamard gates (H), phase gates (S), controlled-NOT gates (CNOT), and T gates is first converted to a sequence of $\pi/4$ (Clifford) and $\pi/8$ (non-Clifford) Pauli rotations, represented as $P_\theta := \exp(i\theta P)$, where P is a Pauli operator and θ is a rotation angle (Figure 1). The next step is a procedure called transpilation, in which the Clifford operations are moved through the circuit using commutation rules and eventually removed from the circuit, leaving only $\pi/8$ rotations. This process generally makes the operations less parallelizable, but also reduces the total number of rotations in the circuit. The naïve method for performing this transpilation in [17] takes $\mathcal{O}(m^2)$ time, where m is the length of the circuit, as each Clifford operation needs to be commuted through each $\pi/8$ rotation. However, there is a faster algorithm [13] that employs techniques similar to efficient simulations of Clifford operations that reduces this runtime to $\mathcal{O}(m)$, which for the sake of completeness is described in Appendix A.

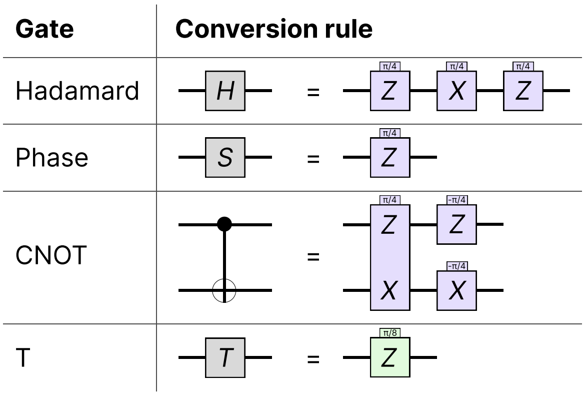


Figure 1 Rules for converting Clifford + T gates into Pauli product rotations. The letters within the boxes on the right (X, Z) represent the Pauli matrix, while the box colour represents the rotation angle, either $\pi/4$ (purple) or $\pi/8$ (green).

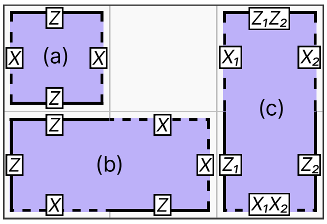


Figure 2 Three types of surface code patches within a 3 × 2 grid of tiles. The edges of the patches represent the Pauli operators X (dashed) and Z (solid). Shown are example (a) single-tile single-qubit, (b) two-tile single-qubit, and (c) two-tile two-qubit patches. Single-qubit patches follow an $XZXZ$ pattern initialized in any position desired, such as (a) and (b). Patches can be extended to multiple tiles using lattice surgery.

After transpilation, all Clifford rotations are removed from the circuit. In our study, the input for the LSSP is a circuit composed of $\pi/8$ rotations and the final qubit measurements. Optionally, $\pi/4$ rotations are also accepted for the scheduling of non-transpiled circuits containing Clifford gates. In our experiments described in Section 5.3, we generate schedules for circuits both before and after the transpilation described in Appendix A, and analyze the challenges and benefits of using this optimization procedure prior to scheduling.

We consider a large array of physical qubits partitioned into patches of surface codes of a desired distance (see Figure 2). Two-qubit patches provide a surface code layout where both qubits can have both X and Z operators accessible from each side of the patch [17]. This way, Y operators can be performed in a single step by connecting the ancilla patch to both X and Z operators simultaneously. Operations like patch initialization, patch measurement, and patch deformation can manipulate qubits associated with these patches [17, 26]. Lattice surgery using ancilla patches enable long-range entangling gates between the logical qubits. We call the set of tiles dedicated to ancilla patch generation the *quantum bus*, with each tile in the quantum bus hosting a *bus qubit*. The qubits associated with those required by the quantum operations are called *data qubits*. Ancilla patches, generated during measurements, can be discarded afterwards, freeing up bus qubits for reuse by newer ancilla patches generated for another operation.

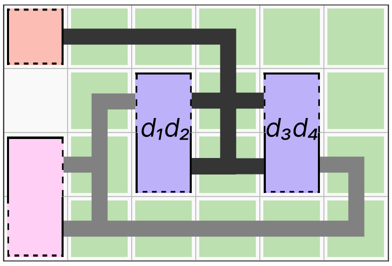


Figure 3 Example of two parallel multi-qubit measurements performed using lattice surgery in a surface code grid with data qubits (shown in purple), bus qubits (green), a magic state storage qubit (red), and an ancillary qubit (pink). The $\pi/4$ rotation corresponds to an $X \otimes I \otimes I \otimes Z$ rotation connected to the ancillary qubit available using eight bus qubits, and the $\pi/8$ rotation corresponds to an $I \otimes Y \otimes Y \otimes I$ rotation connected to the magic state storage qubit available using five bus qubits.

Ancilla patches can be generated in parallel to perform multiple multi-qubit measurements simultaneously, as long as they do not share a bus or a data qubit[1]. Quantum operations involving $\pi/4$ and $\pi/8$ rotations require the entanglement of the qubits required to an extra qubit in a special state [5]. An operation $P_{\pi/4}$ corresponds to a $P \otimes Y$ operation involving an *ancillary qubit* in the zero state. Meanwhile, an operation $P_{\pi/8}$ corresponds to a $P \otimes Z$ operation involving a qubit in a special state called a *magic state.* Figure 3 shows an example of two quantum operations – a $\pi/4$ and a $\pi/8$ rotation – performed in parallel. Operations involving $\pi/8$ rotations may still require an additional corrective $\pi/2$ rotation operation for all qubits originally measured with a probability of 50%, but this correction would take no logical cycles as they are tracked classically [17].

While zero states can be instantly initialized in an ancillary qubit, magic states are prepared through *magic state distillation* [4, 18] which is a costly procedure. Therefore it is customary to assume that this procedure is performed in a separate dedicated zone that interacts with the area comprising the data qubits on which the logical operations are performed. We call this area the *central zone*, which is connected to *magic state storage qubits* located at the boundaries of this zone. Having enough magic state factories providing a distillation rate high enough to meet the magic state consumption rate within the central zone guarantees a continuous supply of magic states to the magic state storage qubits with no overhead required to be taken into account in the LSSP.

Figure 4 illustrates central zones comprising data qubits surrounded by a quantum bus. Magic state storage qubits may be located anywhere at the boundary of a central zone, assuming that magic state distillation is performed externally. Similarly, ancillary qubits are located around the central zone, although this is not a constraint in our models. In Figure 4(a), we show the fast block layout proposed in [17], while in Figure 4(b), we propose a slight modification to Litinski's layout by creating aisles of bus qubits between the data qubit patches and adding a top aisle to the layout to facilitate the parallelization of multi-qubit measurements, as qubits can be connected using multiple paths. Note that, for similar layouts in which multiple qubits are encoded in a number of tiles, Litiski's layout in which two qubits are encoded in two tiles is essentially optimal due to the fact that the number of qubits is related to the size of the boundary.

[1] It is unclear to us whether, with standard lattice surgery operations, a data qubit can contribute to one measurement with an X or Z operator and a second commuting measurement with the other operator simultaneously. Nevertheless, in Section 5.2 we show that scheduling solutions can be improved by about 2% in randomly generated circuits if this case is allowed compared to when it is forbidden.

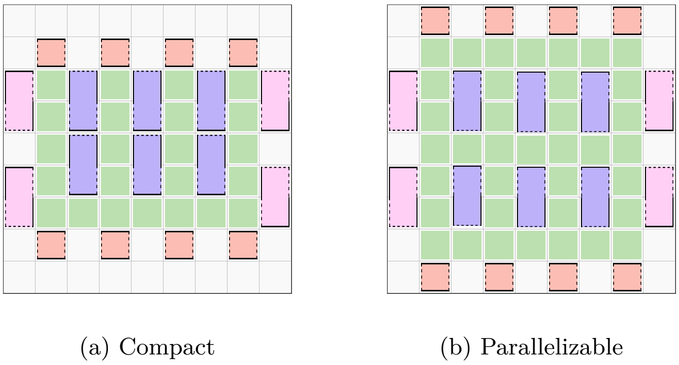

(a) Compact (b) Parallelizable

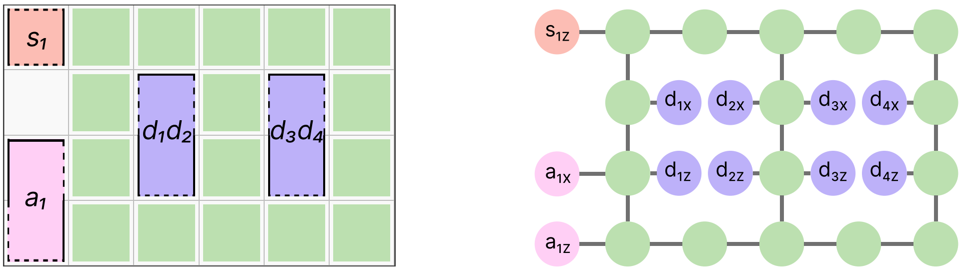
Figure 4 Examples of central zones comprising data qubits surrounded by bus qubits, and with ancillary and magic state storage qubits located at the boundary of the central zone. Given a layout of type (a) with A aisles of data qubits and P data qubit patches in each aisle, a modification can be done to transform it into a layout of type (b) by adding $P(2A+1)$ extra bus qubit tiles. Layout (b) facilitates the parallelization of multi-qubit measurements, as qubits can be connected using multiple paths.

3 The lattice surgery scheduling problem

The LSSP is defined with two necessary inputs: a quantum circuit with Pauli operations and a logical map of the layout. The input quantum circuit is given by a sequence of m Pauli operations $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$ on N qubits considering their order. Each operation $R \in \mathcal{R}$ is characterized by an angle (or measurement) from the set $\{\pm\pi/4, \pm\pi/8, \mathrm{M}\}$, representing $\pi/4$ rotations, $\pi/8$ rotations, and measurement operations, respectively, as well as a Pauli string of length N (e.g., an assignment of a single-qubit Pauli element to each qubit $n \in \{1, 2, \ldots, N\}$). Let $\mathcal{R}^{\pi/4}$ denote the set of $\pm\pi/4$ rotations in $\mathcal{R}$ and $\mathcal{R}^{\pi/8}$ be the $\pm\pi/8$ counterpart. We use R_{in} to represent the single-qubit Pauli operator used by operation R_i on qubit n. Therefore, a rotation R_i requires a qubit n if $R_{in} \neq I$.

In quantum circuits, dependency constraints dictate the order in which quantum operations must be performed. Operations are independent if there is no precedence relationship between them according to the logical constraints of the circuit. Let a dependency check function $c : \mathcal{R} \times \mathcal{R} \to \mathbb{B}$ be used to verify whether a pair of rotations is independent. If $c(R_i, R_j) = 1$ for operations $R_i, R_j \in \mathcal{R}, i \leq j$, then R_i must be completed before starting R_j, that is, R_j depends on R_i. The rules defining the dependency check function are discussed in Section 4.1. Here, it suffices to state that the dependency constraints can be abstracted into a *dependency graph* $\mathcal{G}_{\text{dep}} = (\mathcal{M}, \mathcal{A})$, where the nodes $\mathcal{M}$ represent operations. A directed arc $a_{ij} \in \mathcal{A}$ if $c(R_i, R_j) = 1$.

The LSSP takes a logical map of the surface code layout as another necessary input, which specifies the resources available for running the circuit at the logical level. The layout can be abstracted into an undirected graph $\mathcal{G}_{\text{adj}} = (\mathcal{V}, \mathcal{E})$, called the *adjacency graph*, representing the logical resources, where the vertices $\mathcal{V}$ represent logical qubit patches and the edges $\mathcal{E}$ connect adjacent vertices in the lattice. Vertices are classified according to the type of qubit patch associated with them according to $\mathcal{V} = \{\mathcal{V}^{\text{B}}, \mathcal{V}^{\text{D}}, \mathcal{V}^{\text{S}}, \mathcal{V}^{\text{A}}\}$, where each vertex type is associated with the qubit types bus (B), data (D), magic state storage (S), or ancillary (A). Figure 5 shows an example of the qubit adjacencies represented by an adjacency graph. It should be noted that the assignment of the qubits required by the circuit to the logical data qubits in the hardware should be known before solving the LSSP. Related studies usually



(a) Logical qubit layout.

(b) Adjacency graph.

Figure 5 Example layout for (a) the logical resources in the central zone converted to (b) an adjacency graph. Qubit patches and their accessible Pauli operators are converted into vertices in the adjacency graph, and edges represent the adjacencies between patches. The operators $Z_i Z_j$ (top) and $X_i X_j$ (bottom) of the two-tile, two-qubit patches can be represented by their own vertices, but additional constraints would be required to be added to our models to account for the choice of vertices to use when there is a possibility of connecting the ancilla patch to these operators. We therefore disregard these operators to simplify the generation of the ancilla patches.

randomly assign qubits to patches [2]. However, more-elaborate methods for qubit placement may be used, such as minimizing qubit communication overhead by solving a variant of the quadratic assignment problem [14].

A solution for the scheduling problem is represented by a sequence of time-ordered sets of operations $\mathcal{T} = \{t_1, t_2, \ldots, t_T\}$, where T represents the number of time steps for executing all scheduled operations and each $t \in \mathcal{T}$ is the set of operations scheduled at the respective time step. The duration of a time step is defined by the longest operation scheduled for that time step, measured in logical cycles. As all operations represented by Pauli rotations take one logical cycle, the expected duration of each time step i is equal to one logical cycle regardless of the number of operations scheduled in parallel within i, and the expected number of logical cycles $\mathbb{E}(N)$ to run all time steps is equal to T. The expected number of logical cycles $\mathbb{E}(N)$ can be converted into a concrete time measure, for example, in seconds, by using the wall-clock time of logical cycles.

The LSSP can now be formally stated as follows. Given a dependency graph $\mathcal{G}_{\text{dep}}$ and an adjacency graph $\mathcal{G}_{\text{adj}}$, for any rotation $R \in \mathcal{R}$, the LSSP seeks the time step at which R should be performed to minimize the expected number of total logical cycles $\mathbb{E}(N)$. The LSSP involves making two decisions: sequencing the operations and defining the resources at the logical level needed to perform each operation. We decompose the LSSP based on these decisions, where the sequencing decisions comprise the main problem while the resource usage decisions comprise subproblems to be solved at each time step.

3.1 The primary problem

Let a *pack* be a set of mutually independent logical operations that meet the layout constraints, such as those defined in Section 2. In other words, a pack $p = \widehat{\mathcal{R}} \subseteq \mathcal{R}$, where $c(R_i, R_j) = 0$, $\forall R_i, R_j \in \widehat{\mathcal{R}}$, and p is a valid solution for the subproblem defined in Section 3.2 given the adjacency graph $\mathcal{G}_{\text{adj}}$. Let $\mathcal{P}$ be the collection of all packs. For any pack $p \in \mathcal{P}$, we define a coefficient $A_{ip} = 1$ if rotation $R_i \in p$, or 0 otherwise. The mathematical formulation that we introduce concerns selecting the optimal combination of packs $\mathcal{P}^* \subseteq \mathcal{P}$ such that each operation is covered exactly once. Thus, for any pack $p \in \mathcal{P}$, the decision variable $x_p = 1$ if $p \in \mathcal{P}^*$, or 0 otherwise.

In the LSSP, a pack p_i must be scheduled before another pack p_j if there exists a pair of rotations $R_i \in p_i, R_j \in p_j$ such that $c(R_i, R_j) = 1$. However, if there is also any $c(R_j, R_i) = 1$, then a contradiction exists as this implies that p_j must also be scheduled before p_i. For a set of packs $\mathcal{P}' \subseteq \mathcal{P}$, consider a graph where nodes are the packs in $\mathcal{P}'$ and arcs are the precedence relationships defined. A contradiction exists if a tour in the graph can be found between any subset of packs in $\mathcal{P}'$. Therefore, a feasible solution for the LSSP must not involve such a contradiction. Let us define $\widehat{\mathcal{P}}$ as all collections of packs in $\mathcal{P}$ that have a contradiction. A mathematical formulation for the LSSP is defined as follows:

$$\min \quad \sum_{p \in \mathcal{P}} x_p \tag{1}$$

$$\text{s.t.} \quad \sum_{p \in \mathcal{P}} A_{ip} x_p = 1, \quad \forall R_i \in \mathcal{R}, \tag{2}$$

$$\sum_{p \in \eta} x_p \leq |\eta| - 1, \quad \forall \eta \in \widehat{\mathcal{P}}, \tag{3}$$

$$x_p \in \{0, 1\}, \quad \forall p \in \mathcal{P}. \tag{4}$$

The objective function (1) is defined such that it minimizes the number of packs chosen given that each pack requires the same amount of time to be executed. Constraints (2) impose that each rotation must be scheduled exactly once. Constraints (3), where η refers to packs with a contradiction, ensure that the packs selected will not involve a contradiction by eliminating the formation of tours in the selected packs. Finally, constraints (4) define the domain of the decision variables.

3.2 The subproblem

Defining the set of valid packs $\mathcal{P}$ for the main problem requires solving a packing subproblem. This subproblem involves checking if a set of mutually independent operations can be performed in parallel by verifying whether ancilla patches can be generated on the topological code layout to connect the qubits required by the operations without violating the layout constraints. While checking if a given packing is feasible is enough to define a valid pack for the main problem, we model the packing subproblem such that the usage of logical resources is minimized to reduce the error rate for the operation.

Given $\widehat{\mathcal{R}}$, a set of mutually commuting rotations, where $\widehat{\mathcal{R}}^{\pi/8}, \widehat{\mathcal{R}}^{\pi/4} \subseteq \mathcal{R}'$ represent the subsets of $\pi/8$ and $\pi/4$ rotations, respectively, let $\mathcal{V}_i^{\mathrm{D}}$ be the set of data qubit vertices required by the rotation R_i in the adjacency graph $\mathcal{G}_{\mathrm{adj}} = (\mathcal{V}, \mathcal{E})$. Let the set of incident edges to a vertex $v \in \mathcal{V}$ be defined as $\delta(v) = \{(i,j) \mid i = v \vee j = v, \forall (i,j) \in \mathcal{E}\}$. For any edge $e \in \mathcal{E}$ and rotation $R_i \in \widehat{\mathcal{R}}$, we define a set of decision variables specifying the assignment of the edges as $y_e^i = 1$ if the ancilla patch for R_i uses edge e, or 0 otherwise. Similarly, for any bus qubit $v \in \mathcal{V}^{\mathrm{B}}$, another set of decision variables is defined as $z_v^i = 1$ if R_i uses vertex v, or 0 otherwise. The packing subproblem is then defined as follows:

$$\min \quad \sum_{v \in \mathcal{V}^{\mathrm{B}}} \sum_{R_i \in \widehat{\mathcal{R}}} z_v^i \tag{5}$$

$$\text{s.t.} \quad \sum_{e \in \delta(v)} y_e^i = 1, \quad \forall v \in \mathcal{V}_i^{\mathrm{D}}, R_i \in \widehat{\mathcal{R}}, \tag{6}$$

$$\sum_{e \in \delta(v)} y_e^i = 1, \quad \forall v \in \mathcal{V}^{\mathrm{S}}, R_i \in \widehat{\mathcal{R}}^{\pi/8}, \tag{7}$$

$$\sum_{e \in \delta(v)} y_e^i = 1, \quad \forall v \in \mathcal{V}^{\mathrm{A}}, R_i \in \widehat{\mathcal{R}}^{\pi/4}, \tag{8}$$

$$\sum_{R_i \in \widehat{\mathcal{R}}} z_v^i \leq 1, \quad \forall v \in \mathcal{V}^{\mathrm{B}}, \tag{9}$$

$$2y_{(j,k)}^i \leq z_j^i + z_k^i, \quad \forall (j,k) \in \mathcal{E}, R_i \in \widehat{\mathcal{R}}, \tag{10}$$

$$\sum_{e \in \mathcal{E}} y_e^i = \sum_{v \in \mathcal{V}} z_v^i - 1, \quad \forall R_i \in \widehat{\mathcal{R}}, \tag{11}$$

$$y_e^i, z_v^i \in \{0,1\}, \quad \forall e \in \mathcal{E}, v \in \mathcal{V}, R_i \in \widehat{\mathcal{R}}, \tag{12}$$

The objective function (5) minimizes the total number of bus qubits used to build the ancilla patches. Constraints (6) ensure that all data qubits required by each rotation are connected to the ancilla patch generated for that rotation by a single edge, which guarantees that no data qubit required is crossed by the ancilla patch. Constraints (7) and (8) indicate that there must be exactly one magic state storage qubit and ancillary qubit connected to the $\pi/8$ and $\pi/4$ rotations, respectively. Constraints (9) impose that a bus qubit can be used by no more than one ancilla patch. Constraints (10) ensure the connection between the two variable sets by stating that if the edge (j,k) is used by any ancilla patch, then vertices j and k are also used by it. Constraints (11) guarantee that the ancilla patch generated is a tree. One property of trees is that the number of edges they contain is equal to their number of vertices minus one. The vertices of ancilla patches are composed of the required qubits and all bus qubits used to connect them. Finally, constraints (12) define the domain of the binary decision variables.

Enumerating all $\mathcal{O}(2^m)$ possible packs and solving the packing subproblem for all of them is usually impractical. Although it is possible to address the main problem and the enumeration of the contradicting packs for constraints (3) in real time by using techniques like column generation or cutting-plane algorithms, considering the scale of real quantum circuits, solving the main problem exactly is also impractical. Therefore, we next present an algorithm that schedules operations using a scalable heuristic.

4 A greedy approach to lattice surgery scheduling

To address the LSSP, a crucial step involves building the dependency graph for searching for valid packs. This requires a fast algorithm to perform dependency checks while preserving the true dependency of operations.

Scheduling a long-range multi-qubit operation requires the generation of an ancilla patch connecting the multiple qubits required, called *terminals*. Although any ancilla patch meeting the connectivity requirement is valid, smaller patches are desired as they result in lower error rates. While connecting pairs of qubits is computationally easy, as it requires solving a shortest path problem, connecting to more terminals introduces the NP-hard terminal

Steiner tree problem [16, 6], as the connected qubits are required to be leaf nodes in a tree. Due to the hardness of generating Steiner trees, heuristics are often employed to quickly find near-optimal solutions [22, 25, 7, 21, 19]. Scaling tree generation for potentially millions of operations is essential.

Operations can be scheduled in parallel with the aim of reducing the expected circuit execution time. Checking the feasibility of scheduling multiple operations in parallel poses a challenge, since it requires efficiently packing trees into the adjacency graph. This problem is related to the Steiner forest packing problem, which, along with the Steiner tree packing problem, has been extensively investigated [15, 9, 11, 3, 24].

To tackle these challenges, we have developed fast heuristics for creating the dependency graph, searching for Steiner trees, and packing multiple trees in parallel. The LSSP is solved using an earliest-available-first (EAF) algorithm based on the EAF policy, where operations are scheduled as they become available, given the dependency constraints. We employ a greedy heuristic to solve the forest packing problem, maximizing operations packed among the available candidates. This process is repeated until all operations have been scheduled, considering layout constraints and updated dependency graphs. Algorithm 1 outlines the EAF algorithm for the LSSP, providing a high-level view of our designed approach. Further details on its steps are presented below.

Algorithm 1 Earliest-Available-First Scheduling Algorithm.

```
input Pauli rotation circuit and adjacency graph;
Identify dependencies among operations to build a dependency graph;
while dependency graph is not empty do
    Candidates ← root nodes of the dependency graph;
    Solve the forest packing problem for the candidates, considering the adjacency graph;
    Remove scheduled operations from the dependency graph;
end while
return operations schedule and trees generated for multi-qubit operations
```

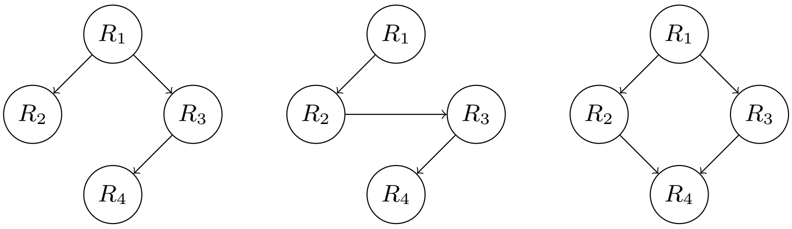
4.1 Dependency graph generation

The order in which operations appear in a quantum circuit determines their dependency relationships; in general, operations must be scheduled following this order. However, the qubits and the Pauli operators required by the operations may allow some operations to commute with others, meaning that they can be applied in any order without affecting the final quantum state. This creates an opportunity to schedule commuting operations in parallel considering their relative positions within the circuit. This section explores different methods for generating dependency constraints, each of which has its own advantages and trade-offs.

As previously stated, the dependency constraints can be abstracted into a dependency graph $\mathcal{G}_{\text{dep}} = (\mathcal{V}, \mathcal{A})$. The vertices, which represent quantum operations, are divided into the subsets $\{\mathcal{V}^{\pi/8}$, $\mathcal{V}^{\pi/4}$, and $\mathcal{V}^{\text{M}}\}$, where each vertex type is associated with a quantum operation type among the $\pi/8$ rotation, the $\pi/4$ rotation, and the qubit measurement (M).

The commutation check for Pauli rotations is defined as follows based on the symplectic representation of Pauli operators as shown in Appendix A. Given two Pauli operators with symplectic representations $P = (\theta_P | x_P | z_P)$ and $Q = (\theta_Q | x_Q | z_Q)$, P and Q commute if $x_P \cdot z_Q + x_Q \cdot z_P \mod 2 = 0$.

A straightforward way to generate the dependency graph following the definition above, which we refer to as the *general rule*, is by visiting each pair of nodes and verifying whether they commute, which requires $\mathcal{O}(|\mathcal{V}|^2)$ commutation checks. Whenever a pair of nodes i and j does not commute, then an arc a_{ij} is added to the graph. The dependency graph



(a) General. (b) Serial. (c) Trivial.

Figure 6 Dependency graphs generated for a circuit with four operations and four qubits, where $R_1 = \{I, X, Y, I\}$, $R_2 = \{Z, I, Z, I\}$, $R_3 = \{I, Y, I, Y\}$, and $R_4 = \{X, X, X, Y\}$. Even though R_1 and R_4 do not commute, the graph for the general rule (a) has only non-redundant arcs. The graph for the serial rule (b) has a single path that respects the order of the operations in the circuit. The graph for the trivial rule (c) restricts commutation and provides a faster way to approximate the general graph.

generated through commutation checks is a directed acyclic graph. Therefore, it can be generated in a transitive reduced form by avoiding redundant arcs that are inferred from the existing arcs in the graph while preserving the relationships of connectivity between nodes. Thus, if $a_{ij}, a_{jk} \in \mathcal{A}$, then a dependency constraint exists between i and k regardless of their commutativity, that is, the dependency check function $c(i, k) = 1$. This graph can be generated in a transitive reduced form using a depth-first search algorithm for the commutation checks. In this way, the number of commutation checks is reduced by $\mathcal{O}(|\mathcal{A}|)$.

Other rules can be employed to generate dependency graphs that impose dependency constraints, even if operations commute. The *serial rule* guarantees that operations are scheduled sequentially and only requires $\mathcal{O}(|\mathcal{V}|)$ operations to be generated by adding the arcs $a_{i(i+1)}$, $\forall i \in \{1, \ldots, |\mathcal{V}| - 1\}$. Another rule, the *trivial rule*, enforces commutation only when qubits required by one operation are disjoint from those required by the other. From the symplectic representation of rotations R_i and R_j, trivial commutation is only possible when $(x_{R_i} \vee z_{R_i}) \cdot (x_{R_j} \vee z_{R_j}) = 0$. We note that if data qubits cannot contribute to different measurements simultaneously, as stated in a footnote in Section 2, the trivial rule results in the true dependency constraints for the scheduling, as operations commuting according to the general rule might violate this condition. Figure 6 shows a comparison of the dependency graphs generated using the rules described for a small circuit. Different rules introduce some trade-offs in terms of parallelization potential, but the trivial rule proves to be scalable for circuits of a size suitable for practical applications, as discussed in Section 5.2.

4.2 Solving forest packing problems

The LSSP is addressed by employing an EAF algorithm that prioritizes the scheduling of operations as early as possible. An operation becomes a candidate for being scheduled at the subsequent time step if it is not dependent on any other operation that has not yet been scheduled. This condition implies that only operations at the root node of the dependency graph are candidates for being scheduled to occur in the same time step. As the dependency graph is a directed acyclic graph, there must be always at least one node that meets the condition described unless all operations have been scheduled or when qubit availability is considered. In the latter case, waiting until all qubits required by at least one root node are available would circumvent this issue.

Given the set of candidate operations and the adjacency graph, the scheduling of the candidates requires solving a forest packing problem if at least one multi-qubit operation is a candidate, where each tree to be packed represents the ancilla patch to be used to generate the multi-qubit entanglements. Algorithm 2 presents the greedy algorithm designed to solve the forest packing problem. For the sake of brevity, we describe the greedy algorithm implemented only for the scheduling of $\pi/8$ rotations, but the steps are applicable for $\pi/4$ rotations.

Algorithm 2 Greedy Forest Packing Algorithm.

```
 1: input set of candidate operations ($\widehat{\mathcal{R}}$) and adjacency graph ($\mathcal{G}_{\text{adj}} = (\mathcal{V}, \mathcal{E})$);
 2: Initialize forest packed $\mathcal{S} \leftarrow \emptyset$;
 3: for all $R_i \in \widehat{\mathcal{R}}$ do
 4:     Temporarily update $\mathcal{V} \leftarrow \mathcal{V}_i^{\text{D}}$;
 5:     Initialize current tree $\mathcal{G}_i = (\mathcal{V}_i^{\text{D}}, \emptyset)$;
 6:     if $|\mathcal{V}_i^{\text{D}}| > 1$ then
 7:         $\mathcal{G}_i \leftarrow$ the solution to the terminal Steiner tree problem connecting terminals $\mathcal{V}_i^{\text{D}}$ in the graph
            $\mathcal{G}_{\text{adj}}$;
 8:     end if
 9:     if $\mathcal{G}_i \neq \emptyset$ then
10:         if $R_i \in \mathcal{V}^{\pi/8}$ then
11:             if $|\mathcal{V}_i^{\text{D}}| > 1$ then
12:                 Replace $\mathcal{G}_i$ by a node $g$ in $\mathcal{G}_{\text{adj}}$ and $s \leftarrow g$ else $s \leftarrow v, v \in \mathcal{V}_i^{\text{D}}$;
13:             end if
14:             $m^* = \arg\min_{m \in \mathcal{V}^{\text{S}}} d(s, m)$;
15:             if $m^* = \emptyset$ then
16:                 Go to line 3;
17:             end if
18:             Update $\mathcal{G}_i$ with the path found to $m^*$;
19:             $\mathcal{V}^{\text{S}} \leftarrow \mathcal{V}^{\text{S}} \setminus \{m^*\}$;
20:         end if
21:         $\mathcal{S} \leftarrow \mathcal{S} \cup (R_i, \mathcal{G}_i)$ and $\mathcal{V} \leftarrow \mathcal{V} \setminus \{\mathcal{V}_i^{\text{B}}\}$;
22:     end if
23: end for
24: return $\mathcal{S}$
```

First, the set of candidate operations $\widehat{\mathcal{R}}$ and the adjacency graph $\mathcal{G}_{\text{adj}} = (\mathcal{V}, \mathcal{E})$ are input (line 1). We initialize the forest packing set $\mathcal{S}$ as empty (line 2). The greedy algorithm tentatively schedules one random operation $R_i \in \widehat{\mathcal{R}}$ at a time (line 3). The data qubits $\mathcal{V}_i^{\text{D}} \subset \mathcal{V}$ required by R_i define the terminals to be used to generate each tree. For each candidate operation, we temporarily remove from $\mathcal{V}$ all data qubits not required by R_i to avoid using them when generating the trees (line 4).

The forest packing problem requires that all trees generated are element-disjoint, meaning that they can share terminal vertices but not internal vertices or edges. Each tree must be a subgraph of $\mathcal{G}_{\text{adj}}$. The tree generation may require two steps. First, if multiple terminals are required, a Steiner tree is generated to connect only the required terminals (lines 6–8). Next, if the candidate is a $\pi/8$ rotation, a vertex associated with a magic state storage qubit is added to the tree (lines 10–20).

The Steiner tree is generated in line 7 by solving the terminal Steiner tree problem to connect the terminals $\mathcal{V}_i^{\text{D}}$ within the graph $\mathcal{G}_{\text{adj}}$. We implement a modified version of Mehlhorn's algorithm [20] for the terminal Steiner tree problem variant. Our algorithm generates a complete graph with all terminal vertices, where each edge represents the shortest path between the connected vertices, which is found using a bidirectional Dijsktra algorithm. Then, it finds the minimum spanning tree using the Kruskal algorithm to connect all vertices in the complete graph. The Steiner tree generated $\mathcal{G}_i$ is provided, connecting all shortest paths chosen for the minimum spanning tree. Since terminals are required to be leaves in the final tree, we ensure that only bus qubits are used as internal vertices in the shortest paths generated by temporarily disconnecting non-bus qubits from the quantum bus. If no feasible tree is found (line 9), then the algorithm moves on to the next candidate, as R_i cannot be scheduled at this time step.

If the candidate R_i is a $\pi/8$ rotation (line 10), one of the magic state storage qubits in $\mathcal{V}^{\mathrm{S}}$ must be added to the tree as a terminal. If this is the case, the magic state storage qubit chosen is the one that is closest to the tree previously generated or to the single qubit required, if this is the case. For the latter, we replace all vertices and edges used for the tree by a single vertex g_i and associate this vertex with the source vertex s. Otherwise, s is associated with the single qubit required by R_i (lines 11–13). Then, we solve a shortest path problem for each magic state storage qubit $m \in \mathcal{V}^{\mathrm{S}}$ from the source s to the magic state storage qubit m. Given the distance $d(s, m)$ between the source s and a magic state storage qubit m, the magic state storage qubit chosen m^* is the one closest to s (line 14); however if no magic state storage qubit can be connected to the tree, this operation is skipped (lines 15–17). If a feasible connection is found, the algorithm connects the shortest path found to the tree (line 18) and removes m^* from the set of storage qubits available (line 19).

Once a tree connecting all terminals is generated, this operation is considered to be scheduled at the current checked time step, by adding the operation R_i and the tree $\mathcal{G}_i$. This process is repeated for another candidate operation using an updated version of the adjacency graph in which all bus qubits $\mathcal{G}_i^{\mathrm{B}}$ used by the set of trees generated in this time step are removed from the graph (line 21). Whenever a tree cannot be generated for a candidate, then the candidate is not scheduled at this time step and waits until the next one. Once all candidates have been checked, the algorithm returns the set of operations scheduled and the trees generated for each of them (line 24).

The scheduling generation is significantly sped up by caching solutions found during the process, such as those for the shortest path problem, for the terminal Steiner tree problem, and for the forest packing problem. Thus, whenever a problem arises for inputs that have already been checked, the previous solution can quickly be retrieved from the cache and checked if it is feasible for the current time step.

5 Computational experiments

This section describes the results of the extensive computational experiments we performed to test the greedy approach proposed to solve the LSSP. The experiments were run using the Google Cloud Platform with nodes comprising an Intel Xeon Gold 6268CL CPU with a 2.80 GHz clock and 256 GB of RAM, limited to a single computing core per run. We implemented the greedy scheduling algorithm in Python and used the NetworkX library for graph operations.

5.1 Test circuits

Two sets of circuits were used to test the implemented algorithms. The first set contains random circuits generated using a structure we defined, while the second contains circuits generated in the Clifford + T basis to represent quantum circuits with real applications. In particular, we used Hamiltonian simulation of various systems as example circuits with real-world applications.

The random circuits were created using a scripted approach to follow the characteristics of circuits that emulate valid outputs of transpilation. An N-qubit circuit consists of many $\pi/8$ rotations requiring a specified number of qubits followed by N final qubit measurements.

Key characteristics considered during circuit generation include the circuit length m, the total number of qubits required N, and the average percentage of qubits involved in each operation $N_{\%}$.

When sampling a rotation for a random circuit, the process begins by determining the number of qubits in its Pauli operator, drawn from a normal distribution $\mathcal{N}(N \times N_{\%}, 2)$. Subsequently, the qubits assigned to a Pauli operator according to the specified number of qubits required are randomly chosen. Then, a Pauli operators among X, Y, and Z is assigned to the chosen qubits. For our computational experiments, the random circuits were generated with combinations from the sets $m = \{10{,}000,\ 20{,}000,\ 30{,}000\}$, $N = \{10, 30, 50\}$, and $N_{\%} = \{0.15, 0.50, 0.85\}$. This totals 27 combinations, where each was generated by five seeds, resulting in 135 random circuits.

The application-inspired circuits followed well-known quantum algorithms for simulating quantum dynamics. They were generated using Hamiltonian simulation for a single step of a Suzuki–Trotter decomposition. Five types of Hamiltonians were generated: electronic structure Hamiltonians of interest in quantum chemistry, one-dimensional chains of random Pauli interactions, transverse-field Sherrington–Kirkpatrick model (TFSK) Hamiltonians, rotated surface code (RSC) Hamiltonians, and one-dimensional Heisenberg XYZ Hamiltonians. For quantum chemistry circuits, we used the electronic structure Hamiltonians of the molecules H_2, LiH, and H_2O. The representations of the Hamiltonians for these molecules were generated using Tangelo [23], an open source Python module for quantum chemistry. We implemented a single step of a Trotter decomposition using PennyLane [1], another open source Python module for quantum algorithms, ensuring an error rate below 10^{-5}. The circuits for one-dimensional chains of random Pauli interactions were generated by randomly selecting Pauli interactions between all sets of three neighbouring qubits, with each interaction having a different random interaction strength between -1 and 1, along with a fixed X-type interaction on each qubit of strength 1, with an evolution time of 1. We considered one-dimensional chains with 10, 20, 30, 40, and 50 qubits. The TFSK Hamiltonian circuits involved X-type interactions of strength 1 on all qubits and ZZ interactions between all pairs of qubits with interaction strength randomly chosen between -1 and 1, and evolved for a time of 1. We explored 10- and 15-qubit circuits. The circuits using RSC Hamiltonians comprise all the check operators of the RSC of distances 3, 5, and 7, with each of the X and Z check operators given a uniformly random weight between 0 and 1, and evolved for a time of 1. The one-dimensional Heisenberg XYZ Hamiltonians consisted of Heisenberg XYZ interactions, with weights for the XX, YY, and ZZ terms randomly chosen using an uniform distribution between -1 and 1, and were evolved for a time of 1. One-dimensional chains with 5, 10, 20, 30, and 40 qubits were considered. All these circuits were decomposed using a Solovay–Kitaev algorithm that decomposes gates with arbitrary rotations into Clifford + T gates, with at L^2-norm error of at most 2.5×10^{-2}. They were then converted to Pauli rotations using the rules shown in Figure 1 and optimized using the transpilation algorithm described in Appendix A. A table summarizing the characteristics of these circuits, before and after the transpilation, is given in Appendix B.

5.2 Analysis of dependency graph generation rules

In this round of experiments, we solved the LSSP using the dependency graph generation rules described in Section 4.1. The layouts considered were generated following the architecture presented in Section 2, with the number of data qubits equal to N and a fixed number of magic state storage qubits $|\mathcal{V}^{\mathrm{S}}| = 3$ located around the central zone. The percentage gap used to compare solutions is defined as $gap = 100(S - S^*)/S^*$, for a solution S compared to another solution S^*.

Table 1 Average statistical values for the experiments performed using each dependency graph generation rule. $\overline{W}$: the average dependency graph width; t_{dep}: the dependency graph generation time, in seconds; t_{sch}: the scheduling time using the earliest-available-first algorithm, in seconds; t_{tot}: the total time ($t_{\text{dep}} + t_{\text{sch}}$); *gap*: the percentage gap of an LSSP solution S for a circuit compared to the best solution among the three rules S^*. Times are not reported for the serial rule, as solutions for the LSSP found after the dependency graph is generated using the serial rule are trivially determined by multiplying the circuit depth by the expected time needed to execute each operation.

Dep. graph generation rule	$\overline{W}$	t_{dep} (s)	t_{sch} (s)	t_{tot} (s)	*gap* (%)
General	1.63	133.4	386.4	519.8	0.00
Trivial	1.06	0.3	340.5	340.8	1.91
Serial	1.00	–	–	–	6.11

Before presenting the results of our experiments, we define a metric to analyze the dependency graphs generated by each rule. Let us denote the depth of a node i in the dependency graph $\mathcal{G}_{\text{dep}} = (\mathcal{N}, \mathcal{A})$ as $D(i) = \max_{(i,j)\in\mathcal{A}}[D(j) + 1]$, where $(i, j) \in \mathcal{A}$ indicates a directed arc from node i to node j. Then, the width W_d of $\mathcal{G}_{\text{dep}}$ at the depth d is defined as the total number of nodes with depth d in $\mathcal{G}_{\text{dep}}$, that is, $W_d = |\{i \in \mathcal{N}|D(i) = d\}|$. Given that $D_{\max} = \max_{i\in\mathcal{N}} D(i)$ is the maximum depth of $\mathcal{G}_{\text{dep}}$, we define the average width of $\mathcal{G}_{\text{dep}}$ as

$$\overline{W} = \frac{1}{D_{\max}} \sum_{d=1}^{D_{\max}} W_d. \tag{13}$$

In other words, if we were to relax the layout constraints, $D(i)$ would denote at which time step the operation i is scheduled, W_d would be the number of operations scheduled for time step d, and $\overline{W}$ would represent the average number of operations scheduled per time step. Based on these definitions, $\max(D) = \max_{i\in\mathcal{R}} D(i)$ represents the circuit depth and is a lower bound for the number of time steps in the optimal solution of the LSSP. In addition, $\overline{W}$ is a measure of the parallelization potential of a circuit. Therefore, for the serial rule (see Section 4.1), $\overline{W} = 1$ and $\max(D) = \mathcal{R}$, that is, the circuit depth is equal to the circuit length.

Table 1 displays the average results obtained in the set of experiments for each rule used to generate the dependency graph. We note that serial scheduling can be considered an upper bound for the LSSP. Complementary data generated from this round of experiments are presented in Table 2 and Figure 7.

The trivial rule is computationally lighter than the general rule, with a shorter average wall-clock time for dependency graph generation and scheduling. Its efficiency increases with the number of rotations, making it ideal for large-scale circuits. Although its solutions are, on average, 1.91% worse than the best known solutions, there is potential for improvement if simultaneous data qubit contributions to multiple measurements are allowed. Serial solutions are, on average, 6.11% worse than the best known solutions, but the gap varies with circuit characteristics. For circuits with N = 10 and $N_{\%} = 0.15$, parallel solutions can be 30–38% better than serial ones. However, as more qubits are requested per operation, optimal solutions tend to align with serial scheduling due to reduced parallelization potential.

Based on these observations, we conclude that the trivial rule offers better scalability for circuits with greater length without having a significant impact on the solution quality. In Appendix B, we show that, even after the transpilation, the Hamiltonian simulation circuits generated can require millions of operations to be scheduled. Consequently, for the remaining experiments, we use the trivial rule as the dependency graph generation rule.

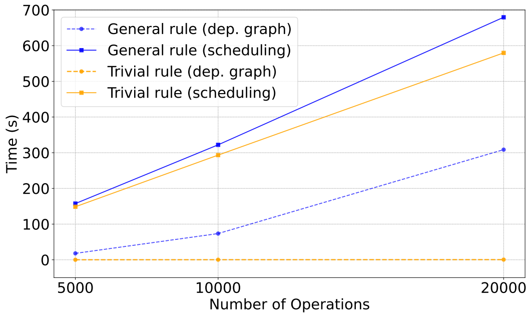


Figure 7 Time needed, in seconds, to generate the dependency graph and the schedule versus circuit size, for the general rule and the trivial rule. The time to generate the dependency graph using the trivial rule does not scale with the number of operations, making it particularly suitable for large-scale circuits.

Table 2 Average gap to the LSSP solution generated using the serial rule. As the total required number of qubits N and the average percentage of qubits required per operation $N_{\%}$ increase, solutions tend to become serialized as the average gap converges to 0%. Therefore, better dependency graph generation rules have no advantage over the serial rule. Conversely, with fewer qubits required per operation, the potential for circuit parallelization increases.

Dep. graph generation rule	$\boldsymbol{N}$	$N_{\%} = 0.15$	$N_{\%} = 0.50$	$N_{\%} = 0.85$
General	10	38.73	2.26	3.98
	30	9.05	0.00	0.01
	50	1.00	0.00	0.00
Trivial	10	30.35	0.84	0.00
	30	4.34	0.00	0.00
	50	0.16	0.00	0.00

5.3 Transpilation analysis

In Table 3, we provide a comparison of solutions for the LSSP on the Hamiltonian simulation circuits using the trivial rule. To capture substantial parallelizability without dramatically increasing space costs, we set $|\mathcal{V}^{\mathrm{S}}| = |\mathcal{V}^{\mathrm{A}}| = \lceil \overline{W} \rceil$ for these runs. The columns with the headings *Pre-transpiled Circuit* and *Post-transpiled Circuit* present the scheduling results for the circuits before and after the transpilation described in Appendix A, respectively. The following are some key observations from the data in the table.

- Our proposed algorithm efficiently solves the LSSP for the Hamiltonian simulation circuits within a reasonable length of time. It can schedule approximately 20,000 operations per second, on average, with a 60% faster performance on post-transpiled circuits compared to pre-transpiled ones.
- Serial scheduling (column *UB*) solutions are improved by 29.5%, on average, when operations are parallelized. This reduction is more pronounced for pre-transpiled circuits (37.3%) compared to post-transpiled ones (21.7%), exemplifying the benefits of parallelization, regardless of transpilation.
- Across all tested circuits, the lower bound of pre-transpiled circuits exceeds the upper bound of post-transpiled ones significantly. This highlights the effectiveness of transpilation in reducing $\mathbb{E}(N)$, contradicting arguments that reduced parallelizability leads to prohibitive runtimes [2]. On average, our experiments demonstrate an 89% reduction in circuit length and an 84% reduction in $\mathbb{E}(N)$ after transpilation.

Table 3 Summary of results for the Hamiltonian simulation circuits. $\mathbb{E}(N)$: the expected number of logical cycles needed to execute the generated schedule; LB: the lower bound for $\mathbb{E}(N)$ given by the circuit depth; UB: the upper bound for $\mathbb{E}(N)$ given by the circuit length; t (s): the total wall-clock time taken to execute the earliest-available-first scheduling algorithm, in seconds.

	Pre-transpiled Circuit				Post-transpiled Circuit			
Circuit	$\mathbb{E}(N)$	LB	UB	t (s)	$\mathbb{E}(N)$	LB	UB	t (s)
H_2	126,874	111,458	155,845	3.5	17,967	16,004	20,190	0.6
LiH	5,996,028	5,969,511	6,165,707	235.2	499,056	498,242	499,956	16.0
H_2O	18,342,657	18,308,449	18,674,983	876.6	1,570,803	1,568,627	1,572,334	54.1
Chain10	161,381	127,456	245,558	34.1	16,209	11,850	26,094	3.7
Chain20	1,480,722	1,210,702	2,439,547	1401.0	178,039	134,262	276,383	43.5
Chain30	2,289,993	1,845,468	3,718,281	4244.8	266,178	205,402	421,462	148.0
Chain40	3,047,568	2,494,727	5,025,943	11764.1	362,395	276,593	568,238	184.3
Chain50	3,787,624	3,128,340	6,286,923	20,774.4	457,277	345,321	712,165	322.3
TFSK10	1,255,636	1,045,161	3,171,274	103.2	356,734	322,515	405,152	9.1
TFSK15	2,235,327	1,630,190	6,916,378	1084.8	793,954	716,148	890,287	23.0
Heisenberg5	1,051,046	989,580	1,451,190	35.4	161,605	145,974	177,390	3.4
Heisenberg10	2,062,007	1,876,235	2,914,850	144.1	309,846	279,904	349,330	7.1
Heisenberg20	4,042,805	3,598,315	5,790,940	1346.1	610,760	523,780	707,142	23.2
Heisenberg30	5,980,488	5,343,285	8,689,920	4735.9	802,016	685,194	1,023,910	25.4
Heisenberg40	7,838,221	7,027,555	11,528,200	13,629.8	1,196,034	1,028,524	1,394,720	56.5
RSC3	83,002	69,610	145,781	4.0	10,332	7800	16,175	0.7
RSC5	712,734	587,120	1,961,200	86.5	143,902	75,726	216,642	31.4
RSC7	1,042,185	804,273	3,707,678	566.0	255,207	130,736	396,707	77.1
Average	3,418,683	3,120,413	4,943,900	3392.8	444,906	387,367	537,460	57.2

6 Conclusion

Our study has investigated the lattice surgery scheduling problem (LSSP), which determines the sequencing of lattice surgery operations on a two-dimensional architecture consisting of topological error correcting codes. A logical layout of the architecture guides resource allocation for quantum operations. Operations requiring multiple qubits require the creation of ancilla patches for the entanglement of the required qubits. We optimize ancilla patches by solving terminal Steiner tree problems to minimize the execution time. Parallel scheduling is investigated by involving ancilla patches representing a quantum bus surrounding data qubits within a dedicated central zone on the layout.

We decompose the LSSP into subproblems based on a forest packing problem. Since enumerating all sets of candidate operations for parallelization is impractical, an algorithm based on the earliest-available-first policy is implemented to select candidates for parallelization, after which a greedy algorithm is used to solve the forest packing problem for the selected operations, taking layout constraints into account.

Our computational experiments reveal that employing a trivial rule to generate dependency constraints enhances scalability for larger circuits. Application-inspired large-scale circuits, comprising up to 18 million quantum gates and 50 qubits, are successfully scheduled within reasonable time frames. We show that parallel scheduling reduces the expected circuit execution time, but it is heavily dependent on the structure of the logical circuit being scheduled. Also, LSSP solutions for optimized circuits outperform scheduling for non-transpiled circuits, reducing the expected number of logical cycles needed to execute the generated schedule by around one order of magnitude in all circuits tested.

There exist cases where transpilation can increase circuit depth, such as in circuits with sequential CNOT gates acting on different qubits followed by sequences of commuting T gates, which would make them lose commutativity after transpilation. However, in general, the removal of Clifford gates is expected to be highly beneficial in reducing FTQC execution time. While our approach considers the scheduling of operations in the Clifford + T gate set, there exist architectures that might improve runtimes using operations in a different basis, such as when arbitrary-angle gates are left in the circuit to be synthesized using resources generated externally to the central zone. In such cases, partial transpilation can still be used to reduce circuit depth in parts of the circuit involving Clifford + T gates.

Finally, the proposed layout considers Y operators to be common in the transpiled circuit. In cases where this is not observed, using more-compact layouts with a smart placement of qubits may be desirable to reduce space costs. A future research direction could be to perform a similar scheduling analysis as was performed in this paper on a colour code lattice, as colour codes allow for easy access to each Pauli measurement, and may lead to more-compact layouts. Our study is a contribution to research on the development of scalable quantum compilers and provides valuable insights into estimating quantum resources required for future fault-tolerant computations.

References

1 J. M. Arrazola, S. Jahangiri, A. Delgado, J. Ceroni, J. Izaac, A. Száva, U. Azad, R. A. Lang, Z. Niu, O. Di Matteo, R. Moyard, J. Soni, M. Schuld, R. A. Vargas-Hernández, T. Tamayo-Mendoza, C. Y.-Y. Lin, A. Aspuru-Guzik, and N. Killoran. Differentiable quantum computational chemistry with Pennylane, 2023. arXiv:2111.09967.

2 M. Beverland, V. Kliuchnikov, and E. Schoute. Surface code compilation via edge-disjoint paths. *PRX Quantum*, 3(2):020342, 2022. doi:10.48550/arXiv.2110.11493.

3 A. Braunstein and A. P. Muntoni. The cavity approach for Steiner trees packing problems. *Journal of Statistical Mechanics: Theory and Experiment*, 2018(12):123401, 2018. doi:10.48550/arXiv.1712.07041.

4 S. Bravyi and J. Haah. Magic-state distillation with low overhead. *Phys. Rev. A*, 86(5):052329, 2012. doi:10.1103/PhysRevA.86.052329.

5 S. Bravyi and A. Kitaev. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Phys. Rev. A*, 71(2):022316, 2005. doi:10.1103/PhysRevA.71.022316.

6 D. E. Drake and S. Hougardy. On approximation algorithms for the terminal Steiner tree problem. *Information Processing Letters*, 89(1):15–18, 2004. doi:10.1016/j.ipl.2003.09.014.

7 M. Fischetti, M. Leitner, I. Ljubić, M. Luipersbeck, M. Monaci, M. Resch, D. Salvagnin, and M. Sinnl. Thinning out Steiner trees: a node-based model for uniform edge costs. *Mathematical Programming Computation*, 9(2):203–229, 2017. doi:10.1007/s12532-016-0111-0.

8 A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A*, 86(3):032324, 2012. doi:10.1103/PhysRevA.86.032324.

9 E. Gassner. The Steiner forest problem revisited. *Journal of Discrete Algorithms*, 8(2):154–163, 2010. doi:10.1016/j.jda.2009.05.002.

10 C. Gidney. Stim: a fast stabilizer circuit simulator. *Quantum*, 5:497, 2021. doi:10.22331/q-2021-07-06-497.

11 N.-D. Hoàng and T. Koch. Steiner tree packing revisited. *Mathematical Methods of Operations Research*, 76:95–123, 2012. doi:10.1007/s00186-012-0391-8.

12 D. Horsman, A. G. Fowler, S. Devitt, and R. V. Meter. Surface code quantum computing by lattice surgery. *New Journal of Physics*, 14(12):123011, 2012. doi:10.1088/1367-2630/14/12/123011.

13 I. H. Kim, Y.-H. Liu, S. Pallister, W. Pol, S. Roberts, and E. Lee. Fault-tolerant resource estimate for quantum chemical simulations: Case study on Li-ion battery electrolyte molecules. *Phys. Rev. Res.*, 4(2):023019, 2022. doi:10.1103/PhysRevResearch.4.023019.

14 L. Lao, B. Van Wee, I. Ashraf, J. Van Someren, N. Khammassi, K. Bertels, and C. G. Almudever. Mapping of lattice surgery-based quantum circuits on surface code architectures. *Quantum Science and Technology*, 4(1):015005, 2018. doi:10.1088/2058-9565/aadd1a.

15 L. C. Lau. Packing Steiner forests. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 362–376. Springer, 2005. doi:10.1007/11496915_27.

16 G. Lin and G. Xue. On the terminal Steiner tree problem. *Information Processing Letters*, 84(2):103–107, 2002. doi:10.1016/S0020-0190(02)00227-2.

17 D. Litinski. A game of surface codes: large-scale quantum computing with lattice surgery. *Quantum*, 3:128, 2019. doi:10.22331/q-2019-03-05-128.

18 D. Litinski. Magic state distillation: not as costly as you think. *Quantum*, 3:205, 2019. doi:10.22331/q-2019-12-02-205.

19 I. Ljubić. Solving Steiner trees: recent advances, challenges, and perspectives. *Networks*, 77(2):177–204, 2021. doi:10.1002/net.22005.

20 K. Mehlhorn. A faster approximation algorithm for the Steiner problem in graphs. *Information Processing Letters*, 27(3):125–128, 1988. doi:10.1016/0020-0190(88)90066-X.

21 T. Pajor, E. Uchoa, and R. F. Werneck. A robust and scalable algorithm for the Steiner problem in graphs. *Mathematical Programming Computation*, 10:69–118, 2018. doi:10.1007/s12532-017-0123-4.

22 C. C. Ribeiro, E. Uchoa, and R. F. Werneck. A hybrid GRASP with perturbations for the Steiner problem in graphs. *INFORMS Journal on Computing*, 14(3):228–246, 2002. doi:10.1287/ijoc.14.3.228.116.

23 V. Senicourt, J. Brown, A. Fleury, R. Day, E. Lloyd, M. P. Coons, K. Bieniasz, L. Huntington, A. J. Garza, S. Matsuura, R. Plesch, T. Yamazaki, and A. Zaribafiyan. Tangelo: An open-source Python package for end-to-end chemistry workflows on quantum computers, 2022. arXiv:2206.12424.

24 Y. Sun, G. Gutin, and X. Zhang. Packing strong subgraph in digraphs. *Discrete Optimization*, 46:100745, 2022. doi:10.1016/j.disopt.2022.100745.

25 E. Uchoa and R. F. Werneck. Fast local search for the Steiner problem in graphs. *ACM Journal of Experimental Algorithmics*, 17:1–22, 2012. doi:10.1145/2133803.2184448.

26 G. Watkins, H. M. Nguyen, K. Watkins, S. Pearce, H.-K. Lau, and A. Paler. A high performance compiler for very large scale surface code computations. *Quantum*, 8:1354, 2024. doi:10.22331/q-2024-05-22-1354.

A Efficient transpilation

The transpilation of Clifford operations out of the circuit works by transforming all of the gates in a circuit into Pauli rotations, and then commuting the Clifford operations past each $\pi/8$ rotation arising from a T gate [17]. The main difficulty with this approach is that the set of rotation gates is not closed under multiplication, even when the rotation gates are restricted to Clifford operations, and since we need to preserve the order of commutation this requires an algorithm reminiscent of bubble sort. Hence, the runtime is $\mathcal{O}(m^2)$, where m is the length of the circuit, since we need to push each Clifford gate past each $\pi/8$ rotation gate individually.

This $\mathcal{O}(m^2)$ algorithm can be avoided, however, if we use the symplectic representation of Clifford gates [13]. Since Clifford gates can be combined, this allows us to perform operations corresponding to multiple rotation commutations in a single step. We perform a simple pass through the circuit, keeping an accumulated Clifford operation representing all of the Clifford operations seen thus far, and at each step either commute this accumulator through a $\pi/8$ gate or combine it with the Clifford gate, depending on the gate present at that particular step. This reduces the runtime of the main optimization step from $\mathcal{O}(m^2)$ to $\mathcal{O}(m)$, leading to massive reductions in computational costs.

A.1 Commuting Clifford and rotation gates

It is important to understand how commuting a Clifford rotation past a $\pi/8$ rotation affects it. Let us assume that U is an n-qubit unitary and P is an n-qubit Pauli operator. Given an arbitrary angle θ, we can then expand the conjugation of the Pauli rotation about P by U as

$$\begin{aligned} U\exp(i\theta P) &= U\exp(i\theta P)U^\dagger U = U\left[\cos(\theta)\mathbb{I} + i\sin(\theta)P\right]U^\dagger U \\ &= \left[\cos(\theta)U\mathbb{I}U^\dagger + i\sin(\theta)UPU^\dagger\right]U = \exp(i\theta UPU^\dagger)U. \end{aligned} \tag{14}$$

Hence, if U is a Clifford operation such that $UPU^\dagger$ is also a Pauli matrix (a defining feature of Clifford operations), commuting U past a Pauli rotation results in U remaining unchanged, while the Pauli operator is updated through conjugation. Since this works for arbitrary angles, it also works in the case of $\pi/8$ rotations.

A.2 Representing Clifford operations

The tableau representation of Clifford gates has been extremely useful in simulating stabilizer circuits and states [10], and is extremely useful in optimizing transpilation [13]. We leverage the ease of representing multiplication of Clifford operators and the conjugation of a given Pauli matrix by Clifford operators. We also note that it is straightforward to transform a given $\pi/4$ rotation into such a representation. Conjugation can be understood as a relatively straightforward combination of rows of the tableau, along with some bookkeeping to keep track of the phase, and multiplication as a sequence of conjugation calls. Further, the initialization of $\pi/4$ rotations can be instantiated through multiplication of Pauli matrices.

For a given n-qubit Pauli operator P, its symplectic representation is a list of $2n+1$ bits, where the first bit corresponds to the phase of the Pauli operator, the next n bits represent the X generators of the Pauli matrix, and the last n bits represent the Z generators of the Pauli matrix. In particular, we say that $P = (\theta|x|z)$, where x and z are each n-bit strings. We can then determine the Pauli element to which P corresponds by inspecting the values of x and z. For a given qubit j, P acts on qubit j as: the identity matrix if x_j and z_j are equal to 0; X if $x_j = 1$ and $z_j = 0$; Y if $x_j = 1$ and $z_j = 1$; and Z if $x_j = 0$ and $z_j = 1$. Note that, since $iXZ = Y$, if both x_j and z_j are equal to 1, we have acquired an additional factor of i that is not accounted for elsewhere in the representation.

A.3 An improved transpilation algorithm

We keep track of the image of a generating set of the Pauli group under conjugation by the Clifford operator C. While the specific order of the generators does not matter, in our representation we alternate between single-qubit X and Z operations acting on qubits of increasing indices. As an example,

$$C = \left(\begin{array}{c|cc|cc} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right)$$

represents a CNOT operation from the first qubit to the second. In this representation, an X generator on the first qubit is mapped to an XX generator when conjugated by this Clifford operator. The improved transpilation algorithm is shown in Algorithm 3. We simply sweep through the circuit, either updating the given gate through conjugation or updating the accumulated Clifford operator, resulting in a runtime that grows linearly with the length of the circuit. In this pseudo-code, for a given rotation R, R_θ denotes the angle of the rotation and R_{Pauli} denotes the Pauli operator of the rotation. We now describe the various update procedures needed on the Clifford tableau representations to implement this algorithm.

Conjugation

With this representation, we can determine the action of conjugation by this Clifford operator on an arbitrary Pauli operator by analyzing the action of each generator making up the given Pauli operator. In particular, if the Pauli operator is given by the symplectic representation $P = (\theta|x|z)$, where x and z are bit strings of length n and θ is a single bit, then there are two main computations we need to determine: to which Pauli operator the operation gets mapped and the overall phase to which to map.

To determine the particular Pauli operator to which P gets mapped under the Clifford operator C, we look at the bits of x and z that are nonzero and perform an XOR operation between the rows of C that correspond to nonzero bits. If we represent the row of C corresponding to a Pauli operator P as C_P, and if C_P^b refers to the all-zeroes string if b is zero or refers to C_P if b is one, we can define the Pauli operators $P_x = C_{X_1}^{x_1} \oplus C_{X_2}^{x_2} \oplus \cdots \oplus C_{X_n}^{x_n} = (\theta_x|x_x|z_x)$ and $P_z = C_{Z_1}^{z_1} \oplus C_{Z_2}^{z_2} \oplus \cdots \oplus C_{Z_n}^{z_n} = (\theta_z|x_z|z_z)$. Then, the Pauli operator to which P is mapped is $P_x \oplus P_z$.

After determining the basis of the given Pauli operator, we must still determine its phase under the mapping. One necessary component is determining the number of commutations that occur when combining each of the C_{X_i} and C_{Z_i}, which can be found by iteratively calculating the z operator for increasing i, and taking the inner product with the x operator of the C_{X_i}.

Another feature that affects the final phase is the result of the factors of i. The number of i's initially in P, the number of i's created when mapping to P_x and P_z, and the number of i's in the representation of $P_x \oplus P_z$ combine to define the final phase of the operator. Specifically, the number of initial i's is given by $n_{i,i} = |x \cdot z|$, the number of intermediate i's is $n_{i,m} = \sum_j |C_{X_j,x} \cdot C_{X_j,z}| + |C_{Z_j,x} \cdot C_{Zj,z}|$, and the number of final i's is given by $n_{i,f} = |(x_x \oplus x_z) \cdot (z_x \oplus z_z)|$. The final change in the phase resulting from the number of i's is then given by $(n_{i,i}+n_{i,m}-n_{i,f})/2 \mod 2 = \theta_i$. Putting all of the above together, the mapping of P under the conjugation of C is given by $CPC^\dagger = (\theta \oplus \theta_x \oplus \theta_z \oplus \theta_c \oplus \theta_i|x_x \oplus x_z|z_x \oplus z_z)$.

Algorithm 3 Efficient Transpilation Algorithm.

```
input Pauli rotation circuit ($\mathcal{R}$);
Let $C$ = tableau($\mathbb{I}$);
Let $\mathcal{R}'$ be an empty set of rotations;
for $R \in \mathcal{R}$ do
    if $R_\theta = \pi/8$ then
        Let $R_{\text{Pauli}} = C$.conjugate($R_{\text{Pauli}}$);
        Append $R$ to $\mathcal{R}'$;
    else if $R_\theta = \pi/4$ or $\pi/2$ then
        Let $C'$ = tableau($R$);
        Let $C = C \times C'$;
    else if $r$ is a measurement then
        Let $R_{\text{Pauli}} = C$.conjugate($R_{\text{Pauli}}$);
        Append $R$ to $\mathcal{R}'$;
    end if
end for
return $\mathcal{R}'$
```

Multiplication

Another attribute needed to implement the improved transpilation algorithm is the ability to multiply Clifford operations. In particular, if U and V are Clifford operations, the mapping of $UVPV^\dagger U^\dagger$ for each generator of the Pauli group must be determined. Fortunately, we already have the mapping $VPV^\dagger$ from the representation of V. To determine the full mapping, we need to determine the conjugation of this operation by U, for which we can use the previous algorithm for conjugation. A given row of UV is then determined by the conjugation of the corresponding row of V by U.

Initialization

The attribute needed is the ability to initialize a Clifford representation from our representation of Pauli rotations, that is, how given $\pi/4$ and $\pi/2$ rotations affect each Pauli generator under conjugation. To understand how a $\pi/2$ rotation about a Pauli operator P affects another Pauli operator Q under conjugation requires an explicit calculation of the commutation according to the equation

$$\begin{aligned} \exp(i\pi/4P)Q\exp(-i\pi/4P) &= (\cos(\pi/4)\mathbb{I} + i\sin(\pi/4)P)\,Q\,(\cos(\pi/4)\mathbb{I} - i\sin(\pi/4)P) \\ &= \frac{1}{2}\left(Q + iPQ - iQP + PQP\right). \end{aligned} \tag{15}$$

From this expression, we can deduce that if P and Q commute, the $\pi/4$ rotation maps Q to itself. Similarly, if P and Q do not commute, then this rotation maps to iPQ.

B Characteristics of the Hamiltonian simulation circuits

Table 4 summarizes the key characteristics of the Hamiltonian simulation circuits generated, as described in Section 5.1, before and after the transpilation described in Appendix A.

Table 4 Summary of the Hamiltonian simulation circuits. "Circuit": the name of the circuit; N: the total number of qubits required by the circuit; $|\mathcal{R}|$: the circuit length (the number of operations in the circuit); $|\mathcal{R}^{\pi/4}|$ and $|\mathcal{R}^{\pi/8}|$: the number of $\pi/4$ and $\pi/8$ rotations in the circuit, respectively; $\overline{W}$: the average dependency graph width (Section 5.2); and $N_\%$: the average number of qubits per operation.

		Pre-transpiled Circuit					Post-transpiled Circuit		
Circuit	N	$\lvert\mathcal{R}\rvert$	$\lvert\mathcal{R}^{\pi/4}\rvert$	$\lvert\mathcal{R}^{\pi/8}\rvert$	$\overline{W}$	$N_\%$	$\lvert\mathcal{R}\rvert$	$\overline{W}$	$N_\%$
H_2	4	155,845	103,441	52,404	1.40	1.00	20,190	1.26	2.31
LiH	12	6,165,707	3,394,011	2,771,696	1.03	1.00	499,956	1.00	8.68
H_2O	14	18,674,983	9,750,953	8,924,030	1.02	1.00	1,572,334	1.00	10.35
Chain10	10	245,558	155,418	90,140	1.93	1.00	26,094	2.20	1.89
Chain20	20	2,439,547	1,522,062	917,485	2.01	1.00	276,383	2.06	2.31
Chain30	30	3,718,281	2,334,375	1,383,906	2.01	1.00	421,462	2.05	2.52
Chain40	40	5,025,943	3,137,085	1,888,858	2.01	1.00	568,238	2.05	2.14
Chain50	50	6,286,923	3,920,154	2,366,769	2.01	1.00	712,165	2.06	2.18
TFSK10	10	3,171,274	2,204,520	966,754	3.03	1.00	405,152	1.26	3.78
TFSK15	15	6,916,378	4,806,264	2,110,114	4.24	1.00	890,287	1.24	4.89
Heisenberg5	5	1,451,190	959,415	491,775	1.47	1.00	177,390	1.22	2.82
Heisenberg10	10	2,914,850	1,935,570	979,280	1.55	1.00	349,330	1.25	2.90
Heisenberg20	20	5,790,940	3,818,520	1,972,420	1.61	1.00	707,142	1.35	2.57
Heisenberg30	30	8,689,920	5,735,250	2,954,670	1.63	1.00	1,023,910	1.49	1.84
Heisenberg40	40	11,528,200	7,563,480	3,964,720	1.64	1.00	1,394,720	1.36	2.49
RSC3	9	145,781	88,299	57,482	2.09	1.00	16,175	2.07	1.98
RSC5	25	1,961,200	1,167,711	793,489	3.34	1.00	216,642	2.86	2.63
RSC7	49	3,707,678	2,166,480	1,541,198	4.61	1.00	396,707	3.03	3.15

Stochastic Error Cancellation in Analog Quantum Simulation

Yiyi Cai
Institute for Quantum Information and Matter, California Institute of Technology,
Pasadena, CA, USA
Department of Electrical Engineering, California Institute of Technology,
Pasadena, CA, USA

Yu Tong
Institute for Quantum Information and Matter, California Institute of Technology,
Pasadena, CA, USA

John Preskill
Institute for Quantum Information and Matter, California Institute of Technology,
Pasadena, CA, USA
AWS Center for Quantum Computing, Pasadena, CA, USA

Abstract

Analog quantum simulation is a promising path towards solving classically intractable problems in many-body physics on near-term quantum devices. However, the presence of noise limits the size of the system and the length of time that can be simulated. In our work, we consider an error model in which the actual Hamiltonian of the simulator differs from the target Hamiltonian we want to simulate by small local perturbations, which are assumed to be random and unbiased. We analyze the error accumulated in observables in this setting and show that, due to stochastic error cancellation, with high probability the error scales as the square root of the number of qubits instead of linearly. We explore the concentration phenomenon of this error as well as its implications for local observables in the thermodynamic limit. Moreover, we show that stochastic error cancellation also manifests in the fidelity between the target state at the end of time-evolution and the actual state we obtain in the presence of noise. This indicates that, to reach a certain fidelity, more noise can be tolerated than implied by the worst-case bound if the noise comes from many statistically independent sources.

2012 ACM Subject Classification Mathematics of computing → Ordinary differential equations; Hardware → Quantum computation; Mathematics of computing → Probability and statistics

Keywords and phrases Analog quantum simulation, error cancellation, concentration of measure

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.2

Funding *Yiyi Cai*: Y.C. acknowledges funding from Arthur R. Adams Summer Undergraduate Research Fellowship.
Yu Tong: Y.T. acknowledges funding from U.S. Department of Energy Office of Science, Office of Advanced Scientific Computing Research (DE-SC0020290), and from U.S. Department of Energy, Office of Science, Basic Energy Sciences, under Award Number DE-SC0019374.
John Preskill: J.P. acknowledges support from the U.S. Department of Energy Office of Science, Office of Advanced Scientific Computing Research (DE-NA0003525, DE-SC0020290), the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Systems Accelerator, and the National Science Foundation (PHY-1733907).The Institute for Quantum Information and Matter is an NSF Physics Frontiers Center.

Acknowledgements We thank Minh Tran, Matthias Caro, Mehdi Soleimanifar, Hsin-Yuan Huang, Andreas Elben, ChunJun Cao, and Christopher Pattison for helpful discussions. We thank Manuel Endres for pointing us to Ref. [40].

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 2; pp. 2:1–2:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Quantum computers are expected to outperform classical computers at solving certain problems of interest in physics, chemistry, and materials science. Simulating the dynamics of many-body quantum systems is an especially hard problem for classical computers, making quantum dynamics a particularly promising arena for seeking quantum advantage. Eventually, scalable fault-tolerant quantum computers will be able to perform accurate simulations of quantum dynamics, but these robust large-scale quantum machines are not likely to be available for many years. Meanwhile, what are the prospects for reaching quantum advantage using near-term quantum simulators that are not error-corrected?

Circuit-based quantum algorithms for quantum simulation offer great flexibility and can be error-corrected, but with current quantum technology analog quantum simulators may offer substantial advantage in the system size and time that can be achieved in simulation [15,17,35]. Analog quantum processors have tunable Hamiltonians running on quantum platforms, but need not have universal local control to perform informative simulations [15, 18, 20]. However, because these devices are not error-corrected, it is especially important to understand how errors accumulate during analog simulations of quantum dynamics.

Recently Trivedi et al. used the Lieb-Robinson bound to show that the errors in expectation values of local observables can be independent of system size for short time evolution [47]. They used an error model in which the actual Hamiltonian realized in the device differs from the desired target Hamiltonian by small local perturbations. More precisely, they considered a geometrically local Hamiltonian on a d-dimensional lattice with N sites (each occupied by a qubit), and assumed that the actual Hamiltonian H' and the target Hamiltonian H are related through

$$H' = H + \delta \sum_{i=1}^{M} V_i. \tag{1}$$

Here each V_i is a local term with $\|V_i\| \leq 1$, $M = \mathcal{O}(N)$ denotes the number of independent error terms, and δ is a small number characterizing the magnitude of the local perturbations. One of their main conclusions is that the error in the expectation value of a local observable at time t is at most $\mathcal{O}(t^{d+1}\delta)$, where t^{d+1} is essentially the volume of the local observable's Lieb-Robinson past light cone, and is independent of the system size. For a general observable that is not necessarily local, or for t large enough so that information has the time to reach every part of the system, the error is at most $\mathcal{O}(Nt\delta)$ as expected from first-order perturbation theory.

This result can be seen as a worst-case bound, which applies even if the small local perturbations are chosen adversarially to produce the largest possible error. However, this worst-case choice is unlikely to occur in practice. For estimating the accumulated error that should be anticipated under realistic conditions, it is often beneficial to consider a probabilistic error model rather than an adversarial one. To be concrete, we consider the error model

$$H' = H + \sum_{i=1}^{M} g_i V_i. \tag{2}$$

where in contrast to (1), we assume that the local perturbations are stochastic and statistically independent, e.g., each g_i is an independent Gaussian random variable with mean 0 and standard deviation δ. Instead of the worst case, we may now consider the accumulation of error in the average case. That is, we envision sampling H' from an ensemble of possible

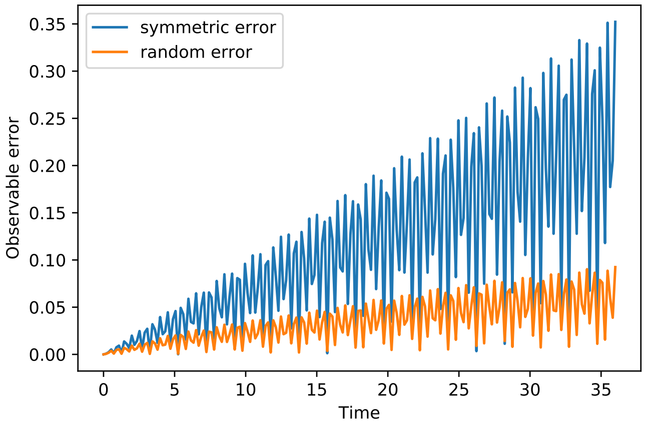


Figure 1 Numerical study of observable error of a 18-qubit system time-evolving under a Heisenberg XXX Hamiltonian $H = -J\sum_i \vec{\sigma}_i \cdot \vec{\sigma}_{i+1} + \sum_i (h+g_i)X_i$. Local observable Y_2 is measured on the second qubit at time t, with the system initial state being $|0\rangle^{\otimes N}$, and the local perturbations are $V_i = X_i$. The coupling strength J between qubits is set to be $(0.2)\pi$, and $h = 2\pi$. We compare the case of $g_i = \delta = 0.01$ (symmetric error), with a Gaussian model of random local perturbations, where $g_i \sim \mathcal{N}(0,\delta^2)$ (random error). The rapid oscillations of the error as a function of time are discussed in Appendix A.

Hamiltonians that might be realized in the device, estimating the error that is typical for this ensemble. In other scenarios, for example in the analysis of the Trotter error in digital quantum simulations [10, 48], the average-case error is found to be much better than the worst case, and the same can be expected in this analog setting. As an example, Fig. 1 shows the difference between the worst-case and average-case error accumulation for time evolution in a one-dimensional Heisenberg spin system perturbed by a site-dependent magnetic field.

Simple classical reasoning provides an intuitive understanding of this finding. The cumulative effect of M error sources, each contributing a Gaussian error with standard deviation δ and mean 0, produces a total Gaussian error with mean 0 and standard deviation $\sqrt{M}\delta$. For $M \gg 1$, this $\mathcal{O}(\sqrt{M})$ cumulative error, resulting from stochastic error cancellations, is significantly suppressed compared to the $\mathcal{O}(M)$ cumulative error which would occur in the absence of such cancellations. Because of the stochastic cancellations, we can tolerate more hardware error (larger δ) than the worst-case error bound suggests.

In this paper, we explore the role of such error cancellations in analog quantum simulators and show that with high probability an error bound with square-root dependence on the system size N can be achieved for general observables, in contrast to the linear N-dependence for the worst-case error bound. That is, the error bound is improved from $\mathcal{O}(Nt\delta)$ to $\mathcal{O}(\sqrt{N}t\delta)$. From this result, we derive an improved bound for local observables in the thermodynamic limit as well. Using the Lieb-Robinson bound, we show that the average-case error of local observables in the thermodynamic limit is bounded above by $\mathcal{O}(t^{d/2+1}\delta)$ as opposed to the $\mathcal{O}(t^{d+1}\delta)$ bound on the worst-case error. For fidelity, we show that the fidelity decays as $\exp(-\mathcal{O}(\sqrt{N}\delta t))$ for small δ as N increases, as observed in [40], and is therefore slower than the exponential decay one would expect from the worst-case bound.

We are only aware of a few works besides [47] that analyze the error in analog simulation. In [34] the error is analyzed by averaging over Haar random states, while in [37] the leading-order error in a Gibbs state is expressed in terms of Fisher information. In contrast, we study how errors accumulate during time evolution of a quantum state. The $\exp(-\mathcal{O}(\sqrt{N}\delta t))$ decay of fidelity is observed in [40] both experimentally and numerically, though in a setting different from ours. In [40], the dominant noise comes from the variation of the Rabi frequency, which is described by a single random variable. Using additional randomness introduced through the eigenstate thermalization hypothesis (ETH) [16] and neglecting oscillatory contributions, the authors are able to provide an explanation of the non-exponential decay. In our setting, we do not assume ETH or neglect any error term, but model the noise as coming from multiple statistically independent sources, and obtain a similar non-exponential decay of fidelity.

2 Main results

Following (2), we consider a generalized setup of the local perturbation model, where each g_i in (2) is a χ-deformed Gaussian defined as follows:

▶ **Definition 1** (χ-deformed Gaussian random variable). *A random variable g is a χ-deformed Gaussian random variable if there exists $\theta \sim \mathcal{N}(0,1)$ such that $g = \chi(\theta)$, and $\chi : \mathbb{R} \to \mathbb{R}$ is a strictly monotonic increasing differentiable function satisfying*

$$\begin{aligned}
&|\mathrm{d}\chi(\theta)/\mathrm{d}\theta| \leq \delta, \\
&\chi(0) = 0, \\
&\chi(+\infty) = \Gamma, \ \ \chi(-\infty) = -\Gamma, \\
&\mathbb{E}[\chi(\theta)] = 0.
\end{aligned} \tag{3}$$

Such a random variable g has the nice properties that $|g| \leq \Gamma$ with probability 1, and $|g| \leq \delta|\theta|$. We allow choosing $\Gamma = +\infty$. This definition helps us generalize beyond the Gaussian noise model. Notably, we have the following examples that can be obtained as χ-deformed Gaussian: (1) the uniform distribution $g_i \sim \mathcal{U}([-\delta', \delta'])$ where $\delta = \sqrt{2/\pi}\delta'$, $\chi(\theta) = \delta' \operatorname{erf}(\theta/\sqrt{2})$, and $\Gamma = \delta'$; (2) the truncated Gaussian distribution for which g_i obeys the Gaussian distribution $\mathcal{N}(0, \delta'^2)$ conditional on $|g_i| \leq \Gamma$, where $\delta = \delta'$ (one can in fact choose δ to be slightly smaller than δ'), and $\chi(\theta) = \operatorname{erf}^{-1}(\operatorname{erf}(\frac{\theta}{\sqrt{2}})\operatorname{erf}(\frac{\gamma}{\sqrt{2}\delta'}))\sqrt{2}\delta'$.

Denoting H as a target Hamiltonian to be simulated and H' as the actual Hamiltonian implemented, we show in Section 3 and 4 that

▶ **Theorem 2.** *On a lattice consisting of N-sites, for Hamiltonians H and H' related through (2) ($M = \mathcal{O}(N)$), with each g_i being an independent χ-deformed Gaussian with $|\mathrm{d}\chi(\theta)/\mathrm{d}\theta| \leq \delta$, $\Gamma \in (0, +\infty]$ (as defined in Definition 1), and $\sqrt{N}t\delta \leq \mathcal{O}(1)$, we have*

$$\left|\operatorname{tr}[\rho e^{iH't} O e^{-iH't}] - \operatorname{tr}[\rho e^{iHt} O e^{-iHt}]\right| \leq \mathcal{O}(a\sqrt{N}t\delta\|O\|) + \mathcal{O}(Nt^2\delta^2\|O\|) \tag{4}$$

with probability $1 - 2e^{-ca^2}$, for arbitrary $a > 0$ and some absolute constant $c > 0$.

Note that Theorem 2 holds even when $\Gamma = \infty$. We assume $\sqrt{N}t\delta \leq \mathcal{O}(1)$, which gives the time scale in which the simulation provides meaningful results. This result is stronger than the $\mathcal{O}(Nt\delta\|O\|)$ scaling one would get without error cancellation, i.e. $g_i = \delta$. This indicates that for given system size N and time t, we can tolerate higher local perturbations up to $\frac{1}{\sqrt{N}t}$ instead of $\frac{1}{Nt}$.

Additionally, one may be interested in the thermodynamic limit ($N \to \infty$) as opposed to a finite system [3] and explore quantum simulation tasks that are stable against extensive errors. More precisely, for a local observable O that is supported only on a constant number of sites, and a geometrically local Hamiltonian H, we want the error bound to be independent of the size of the system. With such an error bound, computing the expectation value of local observables in time evolution falls into the category of "stable quantum simulation tasks" as defined in [47, Prop. 4].

An system-size independent error bound implies that the hardware error (δ) does not need to be scaled down with system size, which is highly desirable for analog simulators. Specifically, we consider a bound for local observables acting on $\mathcal{O}(1)$ adjacent sites in a quantum system on a lattice $\mathbb{Z}_L^d$, where d is the lattice dimension and L is the number of sites in each direction. Combining Theorem 2 with the Lieb-Robinson bound [28], we show in Section 5 that the stability of the quantum task can be stated:

▶ **Theorem 3.** *We consider a geometrically local Hamiltonian H on a d-dimensional lattice $\mathbb{Z}_L^d$ with L sites in each direction, and a local observable O supported on $\mathcal{O}(1)$ sites. The Hamiltonian can be written as $H = \sum_{\alpha\in\mathbb{Z}_L^d} H_\alpha$, where $\|H_\alpha\| = \mathcal{O}(1)$ and H_α acts non-trivially only on sites that are within distance r_0 from α, with $r_0 = \mathcal{O}(1)$. For a H' related to H through* (2) *($M = \mathcal{O}(N)$), with each g_i being an independent χ-deformed Gaussian with $|\mathrm{d}\chi(\theta)/\mathrm{d}\theta| \le \delta$, $\Gamma = \mathcal{O}(1)$ (as defined in Definition 1), $t^{d/2+1}\delta \le \mathcal{O}(1)$, and each site being acted on by only $\mathcal{O}(1)$ of the error terms V_i, we have*

$$\left|\mathrm{tr}[\rho e^{iH't}Oe^{-iH't}] - \mathrm{tr}[\rho e^{iHt}Oe^{-iHt}]\right| = \mathcal{O}\left(at^{\frac{d}{2}+1}\delta\|O\|\right) + \mathcal{O}\left(at\delta\log^{d/2}(\delta^{-1})\|O\|\right) \quad (5)$$

with probability $1 - 2e^{-ca^2}$, for any $a > 0$ and some absolute constant $c > 0$.

This is a stronger bound than the previously established one without error cancellation with leading term of $\mathcal{O}\left(t^{d+1}\delta\right)$ [47]. Note that in the above theorem we require that $\Gamma = \mathcal{O}(1)$, as opposed to $\Gamma \in (0,\infty]$ in Theorem 2. This is to ensure that the Lieb-Robinson bound can be applied to the Hamiltonian H'. $\Gamma = \mathcal{O}(1)$ is physically justifiable because in realistic systems we do not expect to encounter an error that can be arbitrarily large. For the fidelity decay, we have the following theorem:

▶ **Theorem 4.** *On a lattice consisting of N-sites, for Hamiltonians H and H' related through* (2) *($M = \mathcal{O}(N)$), with each g_i being an independent χ-deformed Gaussian with $|\mathrm{d}\chi(\theta)/\mathrm{d}\theta| \le \delta$, $\Gamma \in (0,+\infty]$ (as defined in Definition 1), and $a\sqrt{N}t\delta \le \Delta$, where Δ is a constant that is independent of a, N, t, the fidelity*

$$F = |\langle\phi(t)|\phi'(t)\rangle|^2,$$

where $|\phi(t)\rangle = e^{-iHt}|\phi_0\rangle$, $|\phi'(t)\rangle = e^{-iH't}|\phi_0\rangle$, for initial state $|\phi_0\rangle$, satisfies

$$F \ge e^{-\mathcal{O}(a\sqrt{N}\delta t) - \mathcal{O}(N\delta^2 t^2)} \quad (6)$$

with probability $1 - 2e^{-ca^2}$, for arbitrary $a > 0$ and some absolute constant $c > 0$.

We can see that up to leading order in δ, the fidelity decays exponentially in $\sqrt{N}$ rather than N, thus showing a non-exponential decay of fidelity. From this we can see that in order to make the fidelity bounded away from 0 by a constant, it suffices to have $\delta = \mathcal{O}(1/(\sqrt{N}t))$, rather than $\mathcal{O}(1/(Nt))$ that one would have with a worst-case bound. We will prove this theorem in Section 6.

3 The average error from random noise

We first consider the average observable error accumulated during time evolution and bound

$$\left|\mathbb{E}_{\{g_i\}}[\mathrm{tr}[\rho O'(t)]] - \mathrm{tr}[\rho O(t)]\right| \quad (7)$$

with the notation

$$O(t)=e^{iHt}Oe^{-iHt},\quad O'(t)=e^{iH't}Oe^{-iH't},\quad \rho(t)=e^{-iHt}\rho e^{iHt},\quad \rho'(t)=e^{-iH't}\rho e^{iH't}. \tag{8}$$

We use the evolution under the target Hamiltonian H as a reference frame, and consider the local perturbation in the interaction picture:

$$e^{-iH't}=e^{-iHt}\mathcal{T}e^{-i\int_0^t\sum_i g_iV_i(s)\mathrm{d}s} \tag{9}$$

where $V_i(s)=e^{iHs}V_ie^{-iHs}$ and $\mathcal{T}$ denotes time ordering.

We assume that $\delta\le\mathcal{O}(1/(\sqrt{N}t))$ in the analysis below. Because $M=\mathcal{O}(N)$, we also have $\delta\le\mathcal{O}(1/(\sqrt{M}t))$. We use the Dyson expansion to analyze the accumulation of error:

$$\begin{aligned}&\mathbb{E}[\mathrm{tr}[\rho O'(t)]]-\mathrm{tr}[\rho O(t)]\\&=\sum_{k=1}^{\infty}i^k\int_0^t\mathrm{d}t_1\int_0^{t_1}\mathrm{d}t_2\cdots\int_0^{t_{k-1}}\mathrm{d}t_k\sum_{i_1,\cdots,i_k}\mathbb{E}[g_{i_1}\cdots g_{i_k}]\underbrace{\mathrm{tr}\big[\rho(t)[V_{i_1}(t_1),[\cdots[V_{i_k}(t_k),O]\cdots]]\big]}_{C^{(k)}_{i_1i_2\cdots i_k}}.\end{aligned} \tag{10}$$

With $\|\rho(t)\|_{\mathrm{tr}}\le 1$,[1] Note that $\mathbb{E}[g_{i_1}\cdots g_{i_k}]$ is either 0 (when g_i's do not appear in pairs) or positive (when g_i's appear in pairs), and therefore to upper bound the above quantity in absolute value we only need to upper bound $\big|C^{(k)}_{i_1i_2\cdots i_k}\big|$. Because $||[A,B]||\le\|AB\|+\|BA\|\le 2||AB||$,

$$\big|C^{(k)}_{i_1i_2\cdots i_k}\big|\le||\rho(t)||_{\mathrm{tr}}||[V_{i_1}(t_1),[\cdots[V_{i_k}(t_k),O]\cdots]]||\le 2^k\|O\|. \tag{11}$$

Therefore

$$\begin{aligned}\big|\mathbb{E}[\mathrm{tr}[\rho O'(t)]]-\mathrm{tr}[\rho O(t)]\big|&\le\sum_{k=1}^{\infty}\frac{t^k}{k!}\mathbb{E}\left[\left(\sum_i g_i\right)^k\right]2^k\|O\|\\&=\mathbb{E}[e^{2t\sum_{i=1}^M g_i}-1]\|O\|\\&=\left(\prod_{i=1}^M\mathbb{E}[e^{2tg_i}]-1\right)\|O\|.\end{aligned} \tag{12}$$

Without loss of generality we assume that $\|O\|\le 1$ hereafter. From the above bound we can see that we only need to focus on bounding $\mathbb{E}[e^{2tg_i}]-1$ for each i. Using the fact that $\mathbb{E}[g_i]=\mathbb{E}[\chi(\theta_i)]=0$ from (3) and $|g_i|\le\delta|\theta_i|$, we have

$$\mathbb{E}[e^{2tg_i}]=\sum_{k=0}^{\infty}\frac{(2t)^k}{k!}\mathbb{E}[g_i^k]\le 1+\sum_{k=2}^{\infty}\frac{(2\delta t)^k}{k!}\mathbb{E}[|\theta_i|^k]=\mathbb{E}[e^{2\delta t|\theta_i|}]-2\delta t\mathbb{E}[|\theta_i|]. \tag{13}$$

Using Taylor's theorem in the Lagrange form, with the fact that

$$\frac{\mathrm{d}^k}{\mathrm{d}a^k}\mathbb{E}[e^{a|\theta_i|}]=\mathbb{E}[e^{a|\theta_i|}|\theta_i|^k], \tag{14}$$

we have

$$\mathbb{E}[e^{2\delta t|\theta_i|}]-2\delta t\mathbb{E}[|\theta_i|]\le 1+2\delta^2t^2\mathbb{E}[e^{2\delta t|\theta_i|}|\theta_i|^2]. \tag{15}$$

[1] Here $\|\cdot\|_{\mathrm{tr}}$ denotes the trace norm.

Because $\theta_i \sim \mathcal{N}(0,1)$, for any $a \geq 0$,

$$\mathbb{E}[e^{a|\theta_i|}|\theta_i|^2] = (4a^2e^{a^2})(1+\operatorname{erf} a) + 4\sqrt{\frac{2}{\pi}}ae^{-a^2/2} - \sqrt{\frac{2}{\pi}}ae^{a^2/2} = 1 + \mathcal{O}(a), \tag{16}$$

we have (using $\delta \leq \mathcal{O}(1/(\sqrt{M}t))$)

$$\mathbb{E}[e^{2\delta t|\theta_i|}] - 2\delta t\mathbb{E}[|\theta_i|] \leq 1 + 2\delta^2t^2 + \mathcal{O}(\delta^3t^3) = 1 + 2\delta^2t^2 + \mathcal{O}(M^{-3/2}). \tag{17}$$

By (12), (13), and $\|O\| \leq 1$ we then have

$$\left|\mathbb{E}[\mathrm{tr}[\rho O'(t)]] - \mathrm{tr}[\rho O(t)]\right| \leq \left(1 + 2\delta^2t^2 + \mathcal{O}(M^{-3/2})\right)^M - 1 = 2M\delta^2t^2 + \mathcal{O}(M^{-1/2}). \tag{18}$$

The above derivation leads us to the following theorem:

▶ **Theorem 5** (Average error bound). *On a lattice consisting of N-sites, for Hamiltonians H and H' related through* (2) *($M = \mathcal{O}(N)$), with each g_i being an independent χ-deformed Gaussian with $|\mathrm{d}\chi(\theta)/\mathrm{d}\theta| \leq \delta$, $\Gamma \in (0,+\infty]$, and $\sqrt{N}t\delta \leq \mathcal{O}(1)$, we have*

$$\left|\mathbb{E}_{\{g_i\}}[\mathrm{tr}[\rho O'(t)]] - \mathrm{tr}[\rho O(t)]\right| = \mathcal{O}(N\delta^2t^2\|O\|) \tag{19}$$

This error bound shows that, if we average over multiple instances of the noise, then for the simulation to yield meaningful result up to time t for a system with size N, we need local perturbation to be $\delta = \mathcal{O}(1/(\sqrt{N}t))$, whereas the naive error bound of $\mathcal{O}(Nt\delta)$ would only guarantee a meaningful result only when $\delta = \mathcal{O}(1/(Nt))$. Therefore we can significantly extend the time and system size of the simulation that can be performed with guarantee at the same level of noise.

4 Concentration of the error

In the above section we focused on the expected error, but can the error be significantly larger than its expectation value? This is a question about the concentration of the probability measure, and our main tool is the following lemma:

▶ **Lemma 6** (Gaussian concentration inequality for Lipschitz functions). *Let $f : \mathbb{R}^M \to \mathbb{R}$ be a function which is Lipschitz-continuous with constant* 1 *(i.e. $|f(x) - f(y)| \leq |x - y|$ for all $x, y \in \mathbb{R}^M$), then for any t,*

$$\mathbb{P}\left[|f(X) - \mathbb{E}[f(X)]| \geq t\right] \leq 2\exp(-ct^2) \tag{20}$$

for all $t > 0$ and some absolute constant $c > 0$, where $X \sim \mathcal{N}(0,1)^M$.

The origin of this lemma is rather difficult to find, but its proof can be found at many places, including [41, Theorem 2.1.12] and [7, Chapter 6, Theorem 2.1]. It may appear at first glance that we might need a non-Gaussian version of this result, given that the noise we consider in Definition 1 is not necessarily Gaussian. However, later we will see that a Gaussian version suffices because the noise can be regarded as a function of Gaussian random variables.

Recall that the expectation value $\mathrm{tr}[\rho O'(t)]$ is a function of the noise $\{g_i\}$, which is in turn a function of Gaussian random variables $\{\theta_i\}$ through $g_i = \chi(\theta_i)$. We therefore view $\mathrm{tr}[\rho O'(t)]$ as a function of $\{\theta_i\}$ which we denote by $h(\vec{\theta})$, where $\vec{\theta} = (\theta_1, \theta_2, \cdots, \theta_M)$. Similarly we denote $\vec{g} = (g_1, g_2, \cdots, g_M)$. We will next proceed to obtain a Lipschitz constant for this function. Note that the Lipschitz constant can then be chosen to be the supremum of the 2-norm of the gradient, which we will justify below: applying the mean value theorem (for several variables), for any pair of $\vec{\theta}$ and $\vec{\theta}'$ we have

$$|h(\vec{\theta}) - h(\vec{\theta}')| = |\nabla h(s\vec{\theta} + (1-s)\vec{\theta}') \cdot (\vec{\theta} - \vec{\theta}')|,$$

where $\cdot$ denotes the Euclidean inner product (or the dot product). Therefore

$$|h(\vec{\theta}) - h(\vec{\theta}')| \leq \sup_{\vec{\theta}^*} |\nabla h(\vec{\theta}^*)||\vec{\theta} - \vec{\theta}'|,$$

where the norm $|\cdot|$ on the right-hand side denotes the vector 2-norm, and we have used the Cauchy-Schwarz inequality in arriving at this bound. One can then choose the Lipschitz constant to be anything larger than or equal to $\sup_{\vec{\theta}^*} |\nabla h(\vec{\theta}^*)|$, i.e., any upper bound of $|\nabla h|$, which we will proceed to compute next. We will first bound individual partial derivatives

$$\frac{\partial}{\partial \theta_i} h(\vec{\theta}) = \frac{\mathrm{d}g_i}{\mathrm{d}\theta_i} \partial_{g_i} \mathrm{tr}[Oe^{-iH'(\vec{g})t}\rho e^{iH'(\vec{g})t}], \tag{21}$$

where we make explicit the $\vec{g}$-dependence in H' defined in (2). Because $|\mathrm{d}g_i/\mathrm{d}\theta_i| \leq \delta$ by (3), we only need to bound $\partial_{g_i} \mathrm{tr}[Oe^{-iH'(\vec{g})t}\rho e^{iH'(\vec{g})t}]$:

$$\begin{aligned} |\partial_{g_i} \mathrm{tr}[Oe^{iH'(\vec{g})t}\rho e^{-iH'(\vec{g})t}]| &\leq \|(\partial_{g_i} e^{iH'(\vec{g})t}) O e^{-iH'(\vec{g})t}\| + \|e^{iH'(\vec{g})t} O (\partial_{g_i} e^{-iH'(\vec{g})t})\| \\ &\leq 2\|O\|\|\partial_{g_i} e^{-iH'(\vec{g})t}\| \leq 2\|O\|t. \end{aligned} \tag{22}$$

In the last inequality above we used the fact that

$$\|\partial_{g_i} e^{-iH'(\vec{g})t}\| = \|\int_0^t e^{-iH'(\vec{g})(t-s)} V_i e^{-iH'(\vec{g})s} \mathrm{d}s\| \leq t,$$

where we have used $\|V_i\| \leq 1$. We therefore have $|\partial_{\theta_i} h(\vec{\theta})| \leq 2\|O\|t\delta$. As a result,

$$|\nabla h| = \sqrt{\sum_{i=1}^{M} |\partial_{\theta_i} h|^2} \leq 2\|O\|\sqrt{M}t\delta. \tag{23}$$

Because this holds for all choices of $\vec{\theta}$, the right-hand side is an upper bound of the supremum as well. Therefore we can choose the Lipschitz constant of h to be $C_{\mathrm{Lip}} = 2\|O\|\sqrt{M}t\delta$. $h(\vec{\theta})/C_{\mathrm{Lip}}$ then has Lipschitz constant 1. Through a direct application of Lemma 6, we obtain that for some absolute constant $c > 0$ and any $a > 0$,

$$\mathbb{P}[|h(\vec{\theta}) - \mathbb{E}[h(\vec{\theta})]| \geq aC_{\mathrm{Lip}}] \leq 2e^{-ca^2}. \tag{24}$$

We then have the following result, where we also use $M = \mathcal{O}(N)$:

▶ **Theorem 7** (Concentration bound of observable error). *On a lattice consisting of N-sites, for Hamiltonians H and H' related through* (2) *($M = \mathcal{O}(N)$), with each g_i being an independent χ-deformed Gaussian with $|\mathrm{d}\chi(\theta)/\mathrm{d}\theta| \leq \delta$, $\Gamma \in (0, +\infty]$, and $\sqrt{N}t\delta \leq 1$, we have*

$$\left|\mathrm{tr}[\rho O(t)] - \mathbb{E}[\mathrm{tr}[\rho O'(t)]]\right| \leq 2a\|O\|\sqrt{M}t\delta = \mathcal{O}(a\|O\|\sqrt{N}t\delta) \tag{25}$$

with probability $1 - 2e^{-ca^2}$, for arbitrary $a > 0$ and some absolute constant $c > 0$.

Combining Theorem 5 and Theorem 7, we arrive at the result stated in Theorem 2.

5 Local observables

In this section, we will take locality into consideration to obtain an error bound for local observables that is independent of the system size. Such an error bound is needed to make the simulation meaningful in the thermodynamic limit. We restrict ourselves to spin systems with spatial locality, i.e. systems with Hamiltonians defined on a d-dimensional lattice with N sites in total and L sites in each direction, written as

$$H = \sum_{\alpha \in \mathbb{Z}_L^d} H_\alpha \tag{26}$$

where $\|H_\alpha\| \leq \zeta$ and H_α only acts on spins within a distance r_0 from α, and $r_0 = \mathcal{O}(1)$. A key tool we are going to use is the Lieb-Robinson bound:

▶ **Lemma 8** (Lieb-Robinson Bound, Refs. [9, 24, 47]). *For any local operator O with support S_O, and for any $R > 0$, there exist positive constants u, v that depend only on the lattice such that*

$$\|O(t) - O_R(t)\| \leq \|O\| |S_O| e^{-\mu R} (e^{v\zeta t} - 1) \tag{27}$$

where $O_R(t) = e^{iH_R t} O e^{-iH_R t}$ with $H_R = H - \sum_{\alpha | d(S_{H^\alpha}, S_O) \geq R} H_\alpha$ being the restriction of the Hamiltonian to a region within distance R of S_O.

We can then apply the Lieb-Robinson bound (Lemma 8) to approximate the Heisenberg picture evolution of local observables with that corresponding to the Hamiltonian truncated within their light cones. Specifically, we consider the Heisenberg picture of observable O under the truncated Hamiltonian H_R and H'_R, denoted as:

$$O_R(t) = e^{iH_R t} O e^{-iH_R t}, \quad O'_R(t) = e^{iH'_R t} O e^{-iH'_R t} \tag{28}$$

where we denote H_R as the truncated Hamiltonian acting non-trivially only on sites within distance R from S_O, and H'_R as the Hamiltonian obtained from H' through the same procedure. Assuming $|S_O| \leq \mathcal{O}(1)$, and with $\|H_\alpha\| \leq \zeta$ and $e^{-\mu k} \geq 0$, we arrive at

$$\|O'(t) - O'_R(t)\| \leq \mathcal{O}(\|O\| e^{-\mu R + v\zeta t}), \quad \|O(t) - O_R(t)\| \leq \mathcal{O}(\|O\| e^{-\mu R + v\zeta t}) \tag{29}$$

These bounds then allow us to upper bound the resulting errors in the expectation values through

$$\left|\mathrm{tr}[\rho O'(t)] - \mathrm{tr}[\rho O'_R(t)]\right| \leq \|O'(t) - O'_R(t)\|, \quad \left|\mathrm{tr}[\rho O(t)] - \mathrm{tr}[\rho O_R(t)]\right| \leq \|O(t) - O_R(t)\|,$$

which is true for any quantum state ρ. This is a consequence of the duality between the Schatten 1-norm (the trace distance) and ∞-norm (the spectral norm).

Note that the Lieb-Robinson bound only holds when the strength of local terms does not grow with the size of the system, and this is the reason why we choose $\Gamma = \mathcal{O}(1)$ in Theorem 3, which ensures that all local terms in the Hamiltonian are bounded by a constant that is independent of the system size.

Since now H_R and H'_R acts non-trivially only on $\mathcal{O}(R^d)$, making this the effective system size. We also need to assume that the noise is spread evenly across the whole system, which can be rigorously stated as each site being acted on by only $\mathcal{O}(1)$ of the error terms V_i. Therefore there are only $\mathcal{O}(R^d)$ terms V_i that come into the difference between H_R and H'_R. Consequently we can apply Theorem 2 to get

$$\left|\mathrm{tr}[\rho O'_R(t)] - \mathrm{tr}[\rho O_R(t)]\right| \leq \mathcal{O}(a\sqrt{R^d} t\delta \|O\|) + \mathcal{O}(R^d t^2 \delta^2 \|O\|) \tag{30}$$

with probability $1-2e^{-ca^2}$, for some absolute constant $c>0$ and any $a>0$. Combining the above bounds (29) and (30) together, we obtain

$$\left|\mathrm{tr}[\rho O'(t)]-\mathrm{tr}[\rho O(t)]\right| \leq \mathcal{O}(aR^{d/2}t\delta\|O\|+\|O\|e^{-\mu R+v\zeta t}) \tag{31}$$

Note that if we choose $R=\frac{1}{\mu}(v\zeta t+\log(\delta^{-1}))$, then we get

$$\left|\mathrm{tr}[\rho O'(t)]-\mathrm{tr}[\rho O(t)]\right| \leq \mathcal{O}\left(a\|O\|\left(\frac{v\zeta}{\mu}t+\frac{1}{\mu}\log(\delta^{-1})\right)^{d/2}t\delta+\delta\|O\|\right). \tag{32}$$

Using the fact that $(\alpha+\beta)^d \leq \mathcal{O}(\alpha^d+\beta^d)$ when α,β are constants, we arrive at

$$\left|\mathrm{tr}[\rho O'(t)]-\mathrm{tr}[\rho O(t)]\right| \leq \mathcal{O}\left(at^{\frac{d}{2}+1}\delta\|O\|\right)+\mathcal{O}\left(at\delta\log^{d/2}(\delta^{-1})\|O\|\right) \tag{33}$$

with probability $1-2e^{-ca^2}$. Theorem 3 then follows.

6 Non-exponential fidelity decay

The stochastic error cancellation can also be observed in the fidelity between the target state and the actual state we get at the end of time-evolution, leading to a surprising non-exponential decay of the fidelity for small δ. This is similar to the non-exponential fidelity decay observed in [40]. We consider the fidelity metric as

$$F=|\langle\phi(t)|\phi'(t)\rangle|^2 \tag{34}$$

where

$$|\phi(t)\rangle=e^{-iHt}|\phi_0\rangle, \quad |\phi'(t)\rangle=e^{-iH't}|\phi_0\rangle \tag{35}$$

are pure states with $|\phi(t)\rangle$ denoting the time-evolved state of interest and $|\phi'(t)\rangle$ denoting the state under local perturbation. Here $|\phi_0\rangle$ is the initial state of the system. We will then prove Theorem 4.

Proof of Theorem 4. We are interested in upper-bounding

$$1-F=1-|\langle\phi(t)|\phi'(t)\rangle|^2. \tag{36}$$

We follow similar steps as previous proofs and first bound its expectation value:

$$\begin{aligned}
&\left|\mathbb{E}_{\{g_i\}}[\langle\phi(t)|\phi'(t)\rangle]-1\right| \\
&=\left|\mathbb{E}[\langle\phi_0|e^{iHt}e^{-iH't}|\phi_0\rangle]-1\right| \\
&=\left|\mathbb{E}[\langle\phi_0|\mathcal{T}e^{-i\int_0^t\sum_i g_iV_i(s)\mathrm{d}s}|\phi_0\rangle]-1\right| \\
&=\left|\sum_{k=1}^{\infty}(-i)^k\int_0^t\mathrm{d}t_1\int_0^{t_1}\mathrm{d}t_2\cdots\int_0^{t_{k-1}}\mathrm{d}t_k\sum_{i_1,\cdots,i_k}\mathbb{E}[g_{i_1}\cdots g_{i_k}]\langle\phi_0|V_{i_1}(t_1)V_{i_2}(t_2)\cdots V_{i_k}(t_k)|\phi_0\rangle\right|
\end{aligned} \tag{37}$$

Here, $|\langle\phi_0|V_{i_1}(t_1)V_{i_2}(t_2)\cdots V_{i_k}(t_k)|\phi_0\rangle|\leq 1$ since each local term $\|V_i\|\leq 1$. Note that

$$\sum_{i_1,\cdots,i_k}\mathbb{E}[g_{i_1}\cdots g_{i_k}]=\mathbb{E}\left[\sum_{i_1,\cdots,i_k}g_{i_1}\cdots g_{i_k}\right]=\mathbb{E}\left[\left(\sum_{i=1}^M g_i\right)^k\right]\geq 0.$$

Therefore,

$$\begin{aligned}
\left|\mathbb{E}_{\{g_i\}}[\langle\phi(t)|\phi'(t)\rangle]-1\right| &\le \mathbb{E}\left[\sum_{k=1}^{\infty}\frac{t^k}{k!}\left(\sum_{i=1}^{M}g_i\right)^k\right] \\
&= \mathbb{E}\left[e^{\sum_{i=1}^{M}tg_i}\right]-1 \\
&= \prod_{i=1}^{M}\mathbb{E}\left[e^{tg_i}\right]-1 \\
&\le \prod_{i=1}^{M}(\mathbb{E}[e^{\delta t|\theta_i|}]-\delta t\mathbb{E}[|\theta_i|])-1 \\
&\le \left(1+\delta^2t^2+\mathcal{O}(M^{-3/2})\right)^M-1 \\
&= \mathcal{O}(M\delta^2t^2).
\end{aligned} \tag{38}$$

We can now bound the concentration of the fidelity, i.e, by how much $\langle\phi(t)|\phi'(t)\rangle$ can deviate from its expectation value with large probability. Following the previous setup, we treat $\langle\phi(t)|\phi'(t)\rangle$ as a function of $\{\theta_i\}$, denoted by $h(\vec{\theta})$, where $\vec{\theta}=(\theta_1,\theta_2,...,\theta_M)$. We aim to find a Lipschitz constant for this function. We have

$$|\partial_{g_i}\langle\phi_0|e^{iHt}e^{-iH'(g)t}|\phi_0\rangle| \le \left|\int_0^t\langle\phi_0|e^{iHt}e^{-iH'(g)(t-s)}(-i)V_ie^{-iH'(g)s}|\phi_0\rangle\,\mathrm{d}s\right| \le t \tag{39}$$

and $|\mathrm{d}g_i/\mathrm{d}\theta_i|\le\delta$, giving us $|\partial_{\theta_i}h(\vec{\theta})|\le t\delta$. The Lipschitz constant can then be chosen as

$$\sqrt{\sum_{i=1}^{M}|\partial_{\theta_i}h|^2}\le\sqrt{M}t\delta=C_{\mathrm{Lip}}. \tag{40}$$

$h(\vec{\theta})/C_{\mathrm{Lip}}$ then has Lipschitz constant 1. We then obtain the following bound from Lemma 6:

$$\mathbb{P}[|\langle\phi(t)|\phi'(t)\rangle-\mathbb{E}[\langle\phi(t)|\phi'(t)\rangle]|\ge a\sqrt{M}t\delta]\le 2e^{-ca^2}. \tag{41}$$

for some absolute constant $c>0$ and any $a>0$. Combining the two bounds 38 and (41) derived above, we arrive at

$$|\langle\phi(t)|\phi'(t)\rangle-1|\le a\sqrt{M}\delta t+\mathcal{O}(N\delta^2t^2)$$

with probability $1-2e^{-ca^2}$. This gives us

$$\begin{aligned}
|\langle\phi(t)|\phi'(t)\rangle|^2 &\ge (1-a\sqrt{M}\delta t+\mathcal{O}(N\delta^2t^2))^2 \\
&= 1-2a\sqrt{M}\delta t+\mathcal{O}(N\delta^2t^2) \\
&= e^{-2a\sqrt{M}\delta t-\mathcal{O}(N\delta^2t^2)},
\end{aligned} \tag{42}$$

for $a\sqrt{M}\delta t=\mathcal{O}(1)$. Using $M=\mathcal{O}(N)$, we prove the inequality in the statement of the theorem. ◀

7 Conclusion

In this work, we considered the observable error bounds for analog quantum simulation under random coherent noise coming from independent sources. We showed that such randomness leads to improved scaling in error bounds due to stochastic error cancellation. We studied general observables without locality constraints as well as local observables, finding in both cases that average-case error bounds scale more favorably than worst-case error bounds. Such cancellation indicates a higher tolerance of noise for simulation tasks on near-term analog quantum simulators than suggested by the worst-case bound.

Although our result substantially improves the previous state-of-the-art error bounds, there are still many factors that are not taken into consideration in our analysis. For example, in many-body localized systems [1,2,5,8,19,23,32], our error bound based on the Lieb-Robinson light cone will not be able to capture the slow propagation of information, thus leading to an over-estimation of the error. In general, a tight analysis of the error would require understanding how operators spread in the system, which is a highly non-trivial and system-specific problem [12,33,38,39]. Phenomena such as thermalization should also play an important role, because if a subsystem thermalizes then the error on local operators in the subsystem should no longer accumulate over time. Symmetry has also been shown to be helpful in reducing error in both analog and digital quantum simulations [36,46], and so has randomness in the simulation algorithm and the initial state [4,10,14]. Our results for geometrically local Hamiltonians should be generalizable to the situation with power-law decaying interactions [11,13,22,31,43–45], where the Lieb-Robinson bound is still available when the decay is fast enough. These observations indicate that we may still be able to obtain more accurate characterizations of error accumulation in practical analog simulators.

In this work we focused on quantum systems consisting of qubits or qudits, but many realistic quantum systems involve infinitely many local degrees of freedom and unbounded operators in the Hamiltonian, which makes analysis more difficult [27,42]. We hope to tackle this problem in future works.

Furthermore, we note that an approximate ground-state projection operator can be written as a linear combination of time evolution operators (a fact which is instrumental in the proof of the exponential clustering theorem and 1D area law [6,25,26]) and that approximate ground-state projectors may be used in algorithms for preparing the ground state [21,29,30]. We therefore expect our results to be useful for analyzing how errors in the Hamiltonian affect expectation values of observables in the ground state. We also hope to extend our result to thermal states using the techniques employed in [47].

References

1 Dmitry A. Abanin, Ehud Altman, Immanuel Bloch, and Maksym Serbyn. Colloquium: Many-body localization, thermalization, and entanglement. *Rev. Mod. Phys.*, 91:021001, May 2019. doi:10.1103/RevModPhys.91.021001.

2 Dmitry A. Abanin and Zlatko Papić. Recent progress in many-body localization. *Annalen der Physik*, 529(7):1700169, 2017. doi:10.1002/andp.201700169.

3 Dorit Aharonov and Sandy Irani. Hamiltonian complexity in the thermodynamic limit. In Stefano Leonardi and Anupam Gupta, editors, *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 750–763. ACM, 2022. doi:10.1145/3519935.3520067.

4 Dong An, Di Fang, and Lin Lin. Time-dependent unbounded hamiltonian simulation with vector norm scaling. *Quantum*, 5:459, 2021. doi:10.22331/q-2021-05-26-459.

5 P. W. Anderson. Absence of diffusion in certain random lattices. *Phys. Rev.*, 109:1492–1505, March 1958. `doi:10.1103/PhysRev.109.1492`.

6 Itai Arad, Alexei Kitaev, Zeph Landau, and Umesh Vazirani. An area law and sub-exponential algorithm for 1d systems. *arXiv preprint*, 2013. `arXiv:1301.1162`.

7 Afonso S Bandeira. Lecture notes for mathematics of data science, 2020.

8 D.M. Basko, I.L. Aleiner, and B.L. Altshuler. Metal–insulator transition in a weakly interacting many-electron system with localized single-particle states. *Annals of Physics*, 321(5):1126–1205, 2006. `doi:10.1016/j.aop.2005.11.014`.

9 Sergey Bravyi, Matthew B Hastings, and Frank Verstraete. Lieb-robinson bounds and the generation of correlations and topological quantum order. *Physical review letters*, 97(5):050401, 2006.

10 Chi-Fang Chen and Fernando GSL Brandão. Concentration for trotter error. *arXiv preprint*, 2021. `arXiv:2111.05324`.

11 Chi-Fang Chen and Andrew Lucas. Finite speed of quantum scrambling with long range interactions. *Physical review letters*, 123(25):250605, 2019.

12 Chi-Fang Chen and Andrew Lucas. Operator growth bounds from graph theory. *Communications in Mathematical Physics*, 385(3):1273–1323, 2021.

13 Xiao Chen and Tianci Zhou. Quantum chaos dynamics in long-range power law interaction systems. *Physical Review B*, 100(6):064305, 2019.

14 Andrew M Childs, Aaron Ostrander, and Yuan Su. Faster quantum simulation by randomization. *Quantum*, 3:182, 2019.

15 J. Ignacio Cirac and Peter Zoller. Goals and opportunities in quantum simulation. *Nature Physics*, 8:264–266, 2012.

16 Luca D'Alessio, Yariv Kafri, Anatoli Polkovnikov, and Marcos Rigol. From quantum chaos and eigenstate thermalization to statistical mechanics and thermodynamics. *Advances in Physics*, 65(3):239–362, 2016.

17 Andrew J. Daley, Immanuel Bloch, Christian Kokail, Stuart Flannigan, Natalie Pearson, Matthias Troyer, and Peter Zoller. Practical quantum advantage in quantum simulation. *Nature*, 607(7920):667–676, July 2022.

18 Giacomo De Palma, Milad Marvian, Cambyse Rouzé, and Daniel Stilck França. Limitations of variational quantum algorithms: A quantum optimal transport approach. *PRX Quantum*, 4:010309, January 2023.

19 L. Fleishman and P. W. Anderson. Interactions and the anderson transition. *Phys. Rev. B*, 21:2366–2377, March 1980. `doi:10.1103/PhysRevB.21.2366`.

20 Daniel Stilck França and Raul García-Patrón. Limitations of optimization algorithms on noisy quantum devices. *Nature Physics*, 17(11):1221–1227, October 2021.

21 Yimin Ge, Jordi Tura, and J Ignacio Cirac. Faster ground state preparation and high-precision ground energy estimation with fewer qubits. *Journal of Mathematical Physics*, 60(2), 2019.

22 Zhe-Xuan Gong, Michael Foss-Feig, Spyridon Michalakis, and Alexey V Gorshkov. Persistence of locality in systems with power-law interactions. *Physical review letters*, 113(3):030602, 2014.

23 I. V. Gornyi, A. D. Mirlin, and D. G. Polyakov. Interacting electrons in disordered wires: Anderson localization and low-t transport. *Phys. Rev. Lett.*, 95:206603, November 2005. `doi:10.1103/PhysRevLett.95.206603`.

24 Matthew B Hastings. Lieb-schultz-mattis in higher dimensions. *Physical review b*, 69(10):104431, 2004.

25 Matthew B Hastings. An area law for one-dimensional quantum systems. *Journal of statistical mechanics: theory and experiment*, 2007(08):P08024, 2007.

26 Matthew B Hastings and Tohru Koma. Spectral gap and exponential decay of correlations. *Communications in mathematical physics*, 265:781–804, 2006.

27 Tomotaka Kuwahara, Tan Van Vu, and Keiji Saito. Optimal light cone and digital quantum simulation of interacting bosons. *arXiv preprint*, 2022. `arXiv:2206.14736`.

28 Elliott H. Lieb and Derek W. Robinson. The finite group velocity of quantum spin systems. *Communications in Mathematical Physics*, 28(3):251–257, 1972.

29 Lin Lin and Yu Tong. Near-optimal ground state preparation. *Quantum*, 4:372, 2020. `doi:10.22331/q-2020-12-14-372`.

30 Lin Lin and Yu Tong. Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers. *PRX Quantum*, 3(1):010318, 2022.

31 David J Luitz and Yevgeny Bar Lev. Emergent locality in systems with power-law interactions. *Physical Review A*, 99(1):010105, 2019.

32 Rahul Nandkishore and David A. Huse. Many-body localization and thermalization in quantum statistical mechanics. *Annual Review of Condensed Matter Physics*, 6(1):15–38, 2015. `doi:10.1146/annurev-conmatphys-031214-014726`.

33 Daniel E Parker, Xiangyu Cao, Alexander Avdoshkin, Thomas Scaffidi, and Ehud Altman. A universal operator growth hypothesis. *Physical Review X*, 9(4):041017, 2019.

34 Pablo M Poggi, Nathan K Lysne, Kevin W Kuper, Ivan H Deutsch, and Poul S Jessen. Quantifying the sensitivity to errors in analog quantum simulation. *PRX Quantum*, 1(2):020308, 2020.

35 John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, August 2018.

36 Caleb G Rotello. *Symmetry Protected Subspaces in Quantum Simulations*. PhD thesis, Colorado School of Mines, 2022.

37 Mohan Sarovar, Jun Zhang, and Lishan Zeng. Reliability of analog quantum simulation. *EPJ quantum technology*, 4(1):1–29, 2017.

38 Thomas Schuster, Bryce Kobrin, Ping Gao, Iris Cong, Emil T Khabiboulline, Norbert M Linke, Mikhail D Lukin, Christopher Monroe, Beni Yoshida, and Norman Y Yao. Many-body quantum teleportation via operator spreading in the traversable wormhole protocol. *Physical Review X*, 12(3):031013, 2022.

39 Thomas Schuster and Norman Y Yao. Operator growth in open quantum systems. *arXiv preprint*, 2022. `arXiv:2208.12272`.

40 Adam L Shaw, Zhuo Chen, Joonhee Choi, Daniel K Mark, Pascal Scholl, Ran Finkelstein, Andreas Elben, Soonwon Choi, and Manuel Endres. Benchmarking highly entangled states on a 60-atom analog quantum simulator. *arXiv preprint*, 2023. `arXiv:2308.07914`.

41 Terence Tao. *Topics in random matrix theory*, volume 132. American Mathematical Society, 2023.

42 Yu Tong, Victor V. Albert, Jarrod R. McClean, John Preskill, and Yuan Su. Provably accurate simulation of gauge theories and bosonic systems. *Quantum*, 6:816, 2022. `doi:10.22331/q-2022-09-22-816`.

43 Minh C Tran, Chi-Fang Chen, Adam Ehrenberg, Andrew Y Guo, Abhinav Deshpande, Yifan Hong, Zhe-Xuan Gong, Alexey V Gorshkov, and Andrew Lucas. Hierarchy of linear light cones with long-range interactions. *Physical Review X*, 10(3):031009, 2020.

44 Minh C Tran, Andrew Y Guo, Christopher L Baldwin, Adam Ehrenberg, Alexey V Gorshkov, and Andrew Lucas. Lieb-robinson light cone for power-law interactions. *Physical review letters*, 127(16):160401, 2021.

45 Minh C Tran, Andrew Y Guo, Yuan Su, James R Garrison, Zachary Eldredge, Michael Foss-Feig, Andrew M Childs, and Alexey V Gorshkov. Locality and digital quantum simulation of power-law interactions. *Physical Review X*, 9(3):031006, 2019.

46 Minh C Tran, Yuan Su, Daniel Carney, and Jacob M Taylor. Faster digital quantum simulation by symmetry protection. *PRX Quantum*, 2(1):010323, 2021.

47 Rahul Trivedi, Adrian Franco Rubio, and J Ignacio Cirac. Quantum advantage and stability to errors in analogue quantum simulators. *arXiv preprint*, 2022. `arXiv:2212.04924`.

48 Qi Zhao, You Zhou, Alexander F Shaw, Tongyang Li, and Andrew M Childs. Hamiltonian simulation with random inputs. *Physical Review Letters*, 129(27):270502, 2022.

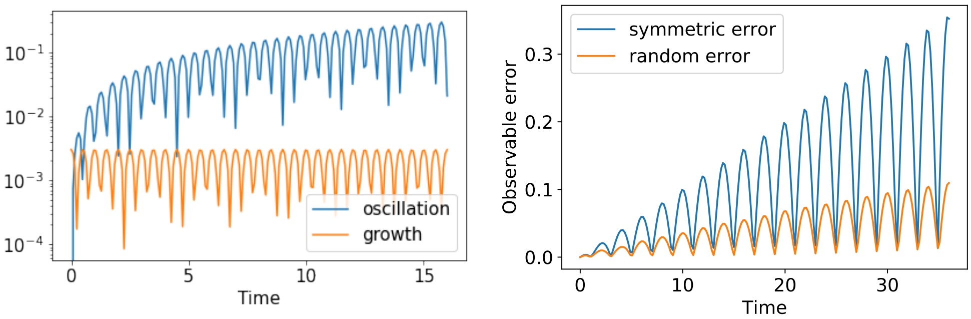


Figure 2 (Left) Comparing the oscillation part $|\langle F(t)\rangle|$ and the growth part $|\langle G(t)\rangle|$ in Eq. (43) that describes the evolution of the error operator. The setup is the same as in Figure 1 except that the system now contains 8 qubits and the observable is $(1/N)\sum_{i=1}^{N} Y_i$. (Right) Error in the observable expectation value for symmetric and random local errors. The simulation is performed with the same parameter setup as in Fig. 1, except with $h = (0.5)\pi$.

A Separation of oscillation and growth in the observable error

In Fig. 1 we observed that the error displays rapid oscillation in time. In this appendix we will investigate the cause of it.

We will examine how the operator O, the operator whose expectation value we want to estimate at the end of the evolution, evolves differently under the target Hamiltonian H and the actual Hamiltonian H'. Using the notation introduced in Eq. (8), we denote by $O(t)$ the time-evolved operator O at time t in the Heisenberg picture under the target Hamiltonian H, and by $O'(t)$ the corresponding operator under the actual Hamiltonian H'. We can write down an equation governing the error $O'(t)-O(t)$, from taking the time derivative in Eq. (8):

$$\frac{\mathrm{d}}{\mathrm{d}t}(O'(t)-O(t)) = \underbrace{i[H, O'(t)-O(t)]}_{F(t)} + \underbrace{i\sum_{i=1}^{M} g_i[V_i, O'(t)]}_{G(t)}. \tag{43}$$

We will show that only the second part $G(t)$ contributes to the growth of the error. Writing down the solution to the above differential equation using Duhammel's principle, for $0 < s < t$ we have

$$O'(t)-O(t) = e^{iH(t-s)}(O'(s)-O(s))e^{-iH(t-s)} + \int_s^t e^{iH(t-u)}G(u)e^{-iH(t-u)}\mathrm{d}u. \tag{44}$$

We observe that if $G(u)=0$ for $s<u<t$, then we would have $\|O'(t)-O(t)\| = \|O'(s)-O(s)\|$, and the error would not grow in magnitude. This shows that $G(t)$ is solely responsible for the growth of the error. The first term on the right-hand side of (43) only rotates $O'(t)-O(t)$.

While $F(t)$ does not contribute to the growth of the error, it nevertheless plays a part in how the derivative changes, as can be seen from (43), which tells us that $\frac{\mathrm{d}}{\mathrm{d}t}\langle O'(t)-O(t)\rangle = \langle F(t)\rangle + \langle G(t)\rangle$. If $|\langle F(t)\rangle| \gg |\langle G(t)\rangle|$, then the error $\langle O'(t)-O(t)\rangle$ will be changing at a rate much faster than its growth, which indicates an oscillatory behavior. We numerically found that this is indeed the case. In Fig. 2, we compare the magnitude of the oscillation part $|\langle F(t)\rangle|$ and the growth part $|\langle G(t)\rangle|$. We can see from the figure that $|\langle F(t)\rangle| \gg |\langle G(t)\rangle|$, which explains the rapid oscillation we see in Fig. 1. In particular, in the parameter setup of Fig. 1, we applied a large X-field whose strength is ten times the coupling constants. This X-field only contributes to $F(t)$ but not $G(t)$, which resulted in $|\langle F(t)\rangle| \gg |\langle G(t)\rangle|$. When we decrease the X-field strength the oscillation frequency decreases accordingly, as can be seen from the right panel of Fig. 2.

Efficient Optimal Control of Open Quantum Systems

Wenhao He
Center for Computational Science and Engineering, MIT, Cambridge, MA, USA
School of Physics, Peking University, Beijing, China

Tongyang Li
Center on Frontiers of Computing Studies, School of Computer Science,
Peking University, Beijing, China

Xiantao Li
Department of Mathematics, Pennsylvania State University, University Park, PA, USA

Zecheng Li
Department of Computer Science and Engineering, Pennsylvania State University,
University Park, PA, USA

Chunhao Wang
Department of Computer Science and Engineering, Pennsylvania State University,
University Park, PA, USA

Ke Wang
Department of Mathematics, Pennsylvania State University, University Park, PA, USA

Abstract

The optimal control problem for open quantum systems can be formulated as a time-dependent Lindbladian that is parameterized by a number of time-dependent control variables. Given an observable and an initial state, the goal is to tune the control variables so that the expected value of some observable with respect to the final state is maximized. In this paper, we present algorithms for solving this optimal control problem efficiently, i.e., having a poly-logarithmic dependency on the system dimension, which is exponentially faster than best-known classical algorithms. Our algorithms are hybrid, consisting of both quantum and classical components. The quantum procedure simulates time-dependent Lindblad evolution that drives the initial state to the final state, and it also provides access to the gradients of the objective function via quantum gradient estimation. The classical procedure uses the gradient information to update the control variables.

At the technical level, we provide the first (to the best of our knowledge) simulation algorithm for time-dependent Lindbladians with an ℓ_1-norm dependence. As an alternative, we also present a simulation algorithm in the interaction picture to improve the algorithm for the cases where the time-independent component of a Lindbladian dominates the time-dependent part. On the classical side, we heavily adapt the state-of-the-art classical optimization analysis to interface with the quantum part of our algorithms. Both the quantum simulation techniques and the classical optimization analyses might be of independent interest.

2012 ACM Subject Classification Theory of computation → Quantum computation theory

Keywords and phrases Quantum algorithm, quantum optimal control, Lindbladian simulation, accelerated gradient descent

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.3

Related Version *Full Version*: `https://arxiv.org/abs/2405.19245` [26]

Funding *Tongyang Li*: National Natural Science Foundation of China (Grant Numbers 62372006 and 92365117); The Fundamental Research Funds for the Central Universities, Peking University.
Xiantao Li: NSF DMS-2111221 and CCF-2312456

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 3; pp. 3:1–3:23
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Zecheng Li: NSF CCF-2312456
Chunhao Wang: NSF CCF-2312456
Ke Wang: NSF DMS-2111221

Acknowledgements We thank the anonymous reviewers for the valuable feedback.

1 Introduction

The ability to control the dynamics of a quantum system to maximize its property has been a persistent pursuit in quantum physics and chemistry [18]. This endeavor has recently gained momentum, spurred by the growing interest in designing quantum information processing devices. One remarkable obstacle in controlling a quantum system's behavior stems from the reality that quantum systems typically evolve in the presence environmental noise. Consequently, the control strategy must take into account system/bath interactions. In the Markovian regime, this problem can be formulated as an optimal control problem based on the Lindblad master equation [38, 24] acting on n qubits,

$$\frac{d}{dt}\rho = \mathcal{L}(t)(\rho) := -i\left[H_0 + \sum_{\beta=1}^{n_{\rm c}} u_\beta(t)\mu_\beta, \rho\right] + \sum_{j=1}^{m}\left(L_j\rho L_j^\dagger - \frac{1}{2}\{L_j^\dagger L_j, \rho\}\right), \tag{1}$$

in conjunction with a control functions $u_\beta(t)$ that enters the system Hamiltonian through the operator μ_β, and we have $n_{\rm c}$ control functions. Here ρ is a density operator on n qubits, and the second term in Eq. (1) is a result of system/bath interactions with L_j's being the jump operators. The quantum optimal control (QOC) is then formulated as an optimization problem following [2]:

$$\max_{\boldsymbol{u}} f[\boldsymbol{u}(t)], \quad f[\boldsymbol{u}(t)] := \operatorname{tr}\big(\mathcal{O}\rho(T)\big) - \alpha\sum_{\beta=1}^{n_{\rm c}}\int_0^T |u_\beta(t)|^2 \mathrm{d}t. \tag{2}$$

The Hermitian operator $\mathcal{O}$ represents the property to be maximized. The term $\boldsymbol{u}(t)$ embodies all the control variables $\{u_\beta\}$ and the last term in the objective function $f[\boldsymbol{u}(t)]$ is regarded as a regularization. It is worthwhile to point out that there are other choices of the objective function [8] in the formulation of the QOC problem. For example, one can guide the Lindblad dynamics (1) toward a target state $\bar{\rho}(T)$. In this case, one can minimize the difference between $\bar{\rho}(T)$ and $\rho(T)$,

$$\min_{\boldsymbol{u}} f[\boldsymbol{u}(t)], \quad f[\boldsymbol{u}] := \|\rho(T) - \bar{\rho}(T)\|^2 + \alpha\sum_{\beta}\int_0^T |u_\beta(t)|^2 \mathrm{d}t. \tag{3}$$

Implicit in both optimization problems Eqs. (2) and (3) is that $\rho(T)$ has to be obtained from the Lindblad equation (1). Thus the main computational challenge comes from the repeated computation of the solution of the Lindblad equation. In this paper, we mainly focus on the optimal control problem with the objective function Eq. (2).

To be able to clearly illustrate the computational complexity, we assume that the Hamiltonian $H(t)$ and the jump operators $L_j(t)$'s are sparse. Moreover, the sparsity structure for each operator does not change over time (i.e., the positions of nonzero entries do not change with time). For a sparse matrix A, we assume we have access to a procedure $\mathcal{P}_A$ that can apply the following oracles:

$$\mathcal{O}_{A,\mathrm{loc}}\left|i,j\right\rangle = \left|i,\nu_A(i,j)\right\rangle, \quad \text{and} \tag{4}$$

$$\mathcal{O}_{A,\mathrm{val}}\left|t,i,j,z\right\rangle = \left|t,i,j,z\oplus A_{i,j}(t)\right\rangle, \tag{5}$$

where $\nu_A(i,j)$ is the index of the j's nonzero entry of column i in A. Particularly for the optimal control problem, we assume we have access to $\mathcal{P}_{H_0}$, $\mathcal{P}_{\mu_\beta}$, and $\mathcal{P}_{L_j}$ for all $j \in [m]$, as well as $\mathcal{P}_{\mathcal{O}}$ for the observable $\mathcal{O}$.

Main contributions

We will present a hybrid quantum/classical algorithm for the QOC problem (1) and (2). The overall algorithm consists of the following elements:

1. A Lindblad simulation algorithm [14, 15, 36] that prepares $\rho(T)$ in a purification form. The complexity of our algorithm exhibits a linear scaling with respect to T with a scaling factor proportional to the L_1 norm of the Lindbladians instead of the $L_{\max}$ norm. The dependence of the complexity on the precision ϵ is only poly-logarithmic. Alternatively, we can also simulate time-dependent Lindbladian using interaction picture [41]. This algorithm applies to important models in experimental physics. For instance, in an ion trap system, it is common to have a time-independent Hamiltonian with norm much larger than the rest of the Lindbladian terms, and thus our algorithm can make the simulation more efficient.
2. The construction of a quantum phase oracle of the gradient of the function f. This is achieved by incorporating the quantum gradient computation algorithms in [21]. This phase oracle will then be interfaced with a classical optimization algorithm.
3. Having approximates of gradients $\nabla f(\boldsymbol{u}(t))$, we use an accelerated gradient descent (AGD) method [27] to solve the optimization problem. In particular, we analyze the influence of the statistical error from the gradient estimation and provide a precise complexity analysis for solving the optimization problem, which essentially characterizes the robustness of AGD for reaching second-order stationary points and may be of independent interest.

In addition to the proposed algorithms, we provide rigorous analysis of the numerical error and precise overall complexity estimates for the hybrid algorithm. Formally, we establish the following result for optimal control of open quantum systems:

▶ **Theorem 1** (main theorem). *Assume there are n_{c} control functions $u_\beta(t) \in \mathrm{C}^2([0,T])$. Further assume*[1] *that $\|H_0\|, \|\mathcal{O}\|, \|\mu_\beta\|, \|L_j\| \leq 1$, and $\alpha \geq 2/T$. There exists a quantum algorithm that, with probability at least $2/3$*[2]*, solves problem (2) by:*

- *reaching a first-order stationary point $\|\nabla f\| < \epsilon$ with (1) using $\widetilde{O}\left(\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T}{\epsilon^{23/8}}\Delta_f\right)$ queries to $\mathcal{P}_{H_0}$ and $\mathcal{P}_{\mu_\beta}$, $\beta = 1, 2, \ldots, n_{\mathrm{c}}$, and $\widetilde{O}\left(mn\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T}{\epsilon^{23/8}}\Delta_f + n\frac{T^{3/2}}{\epsilon^{9/4}}\Delta_f\right)$ additional 1- and 2-qubit gates; or*
- *reaching a second-order stationary point using $\widetilde{O}\left(\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T^{7/4}}{\epsilon^{5}}\Delta_f\right)$ queries to $\mathcal{P}_{H_0}$ and $\mathcal{P}_{\mu_\beta}$, $\beta = 1, 2, \ldots, n_{\mathrm{c}}$ and $\widetilde{O}\left(mn\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T^{7/4}}{\epsilon^{5}}\Delta_f + n\frac{T^{3/2}}{\epsilon^{9/4}}\Delta_f\right)$ additional 1- and 2-qubit gates.*

Here n_{c} and m are respectively the number of control variables and jump operators.

Techniques

Our technical contributions are outlined as follows.

[1] More generally, if $\|H_0\|, \|\mu\| = \Theta(\Lambda)$, it is equivalent to enlarge the time duration T by a factor $O(\Lambda)$.

[2] Using standard techniques, the success probability can be boosted to a constant arbitrarily close to 1 while only introducing a logarithmic factor in the complexity.

- In Section 3, we give efficient quantum algorithms for simulating time-dependent Lindbladians with a scaling factor in time proportional to the L_1-norm of the Lindbladians instead of the $L_{\max}$-norm, as well as poly-logarithmic ϵ dependence. Our simulation algorithm is based on the higher-order series expansion from Duhamel's principle as sketched in [36]. A notable difference from [36] is that in their paper, Gaussian quadratures are used to approximate integrals; however, in our time-dependent case, Gaussian quadratures can no longer be used as, unless upper bounds on the higher-order derivatives of the operators are given in advance. The techniques for obtaining the L_1-norm dependence follow from the rescaling trick in [5], while generalized to Lindbladians. Our time-dependent Lindbladian simulation techniques might be of independent interest.
- In Section 4, we show how to simulate time-dependent Lindbladian using interaction picture [41]. This technique is suited for simulating a Lindbladian $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$ where $\mathcal{L}_1(\cdot) = -i\,[H_1, \cdot]$ is a Hamiltonian with complexity linear in norm of $\mathcal{L}_2$ (up to poly-logarithmic factors) and similar number of simulations of the Hamiltonian H_1 . The simulation scheme is based on a mathematical treatment of the Lindblad equation as a differential equation, and the construction leverages the simulation algorithms shown in Section 3 without rescaling the evolution time. It turns out that using our simulation algorithm in the interaction picture, we obtain better gate complexity compared with directly using the simulation in Section 3 even with the ℓ_1-norm dependence. To the best of our knowledge, this is the first Lindblad simulation algorithm in the interaction picture, which can also be of independent interest.
- In Section 5, we adapt a nonconvex optimization algorithm that can reach first-order stationary points with $\tilde{O}(1/\epsilon^{7/4})$ *noisy* gradient queries with ℓ_2-norm error at most $O(\epsilon^{9/8})$, and reach second-order stationary points with $\tilde{O}(1/\epsilon^{7/4})$ *noisy* gradient queries with ℓ_2-norm error at most $\tilde{O}(\epsilon^3)$. Our setting is different from either gradient descent (GD) or stochastic gradient descent (SGD): Compared to GD we only have access to noisy gradients, while in standard SGD the noise can be adjusted and there is no Lipschitz condition for the noisy gradient. With this novel setting, we successfully design an optimization algorithm based on perturbed accelerated gradient descent (PAGD) [27]. We carefully analyze the error bound in different cases and it turns out that our algorithm reaches an optimal error scaling for PAGD (up to poly-logarithmic factors) in finding a first-order stationary point.

Related work

In addition to the large variety of conventional applications [20], quantum optimal control problems are crucial in near-term quantum computing, because in the architecture of quantum computers, the underlying physical operations such as microwave control pulses and the modulated laser beam can be abstracted as control pulse sequences (see the survey [47] for more detailed discussions), and hence the are inherently quantum control problems. Quantum optimal control also plays a vital role in quantum computing algorithms. For instance, Magann et al. [42] studied the relationship between variational quantum algorithms (VQAs) and quantum optimal control, and showed that the performance of VQAs can be informed by quantum optimal control theory. Banchi and Crooks [4] demonstrated how gradients can be estimated in a hybrid quantum-classical optimization algorithm, and quantum control is used as one important application. In Ref. [40] the authors showed that for a quantum many-body system, if it exists an efficient classical representation, then the optimal control problems on this quantum dynamics can be solved efficiently with finite precision.

There exist heuristic classical methods for solving the quantum optimal control problem, including the monotonically convergence algorithms [48], the Krotov method [45], the GRadient Ascent Pulse Engineering (GRAPE) algorithm [29, 17], the Chopped RAndom-Basis (CRAB) algorithm [9], etc. Furthermore, such heuristics can be extended to quantum optimal control of open quantum systems [31, 32, 46, 23], including [1, 33, 6]. However, these algorithms do not establish provable guarantees for the efficiency of solving the quantum optimal control problem. Meanwhile, the landscape of the quantum control problem has been analyzed in [13, 16, 19], which suggests that for closed quantum systems, the landscape may not involve suboptimal optimizers. However, the implication to the computational complexity still remains open.

Quantum algorithms, due to their natural ability to simulate quantum dynamics, have been developed for quantum control problems [43, 34, 12, 35]. Liu and Lin [39] developed an efficient algorithm to output the integral of the observable in Eq. (2), which can potentially solve a more generalized optimal control problem. These approaches employ hybrid quantum-classical algorithms that combine a quantum algorithm for the time-dependent Schrödinger equation with a classical optimization method. However, these efforts have been focused on closed quantum systems, and quantum control algorithms for open quantum systems require separate techniques.

Open questions

Our paper leaves several open questions for future investigations:

- Are there efficient quantum algorithms for the optimal control of other master equations beyond the Lindbladian equation?
- How to extend the current framework to the control problems with a target density operator $\bar{\rho}(T)$? The challenge in such a control problem (3) is the estimation of the Frobenius norm from the quantum circuit.
- Gaussian quadrature was used in the Lindblad simulation method [36], which significantly suppressed the number of terms in a Dyson-series type of approach, and implies the implementation. The extension of Gaussian quadrature to the current framework with time-dependent Lindbladians would require derivative bounds for the evolution operator from both the drift and jump terms, which is not trivial.

2 Preliminaries

2.1 Notations

For a positive integer m, we use $[m]$ to denote the set $\{1, \dots, m\}$. In this paper, we use two types of notations to denote vectors. For a quantum state, we use the Dirac notation $|\cdot\rangle$ to denote the corresponding state vector. For vectors involved in classical information, e.g., the gradient vector, we use bold font, such as $\boldsymbol{u}$, to denote them. For such a vector $\boldsymbol{u} \in \mathbb{C}^d$, we use subscripts with a norm font to indicate its entries, i.e., $u_1, \dots, u_d$ are the entries of $\boldsymbol{u}$. When we use subscripts with a bold font, such as, $\boldsymbol{u}_1, \dots, \boldsymbol{u}_k$, they are a list of vectors. For a vector $\boldsymbol{v} \in \mathbb{C}^d$, we use $\|\boldsymbol{v}\|$ to denote its *Euclidean norm*. For a matrix $M \in \mathbb{C}^{d\times d}$, we use $\|M\|$ to denote its *spectral norm*, and use $\|M\|_1$ to denote its *trace norm*, i.e., $\|A\|_1 = \mathrm{Tr}(\sqrt{M^\dagger M})$. We also use $[\cdot,\cdot]$ to denote the operator *commutator*, i.e., $[A, B] := AB - BA$, and use $\{\cdot,\cdot\}$ to denote the *anticommutator*, i.e., $\{A, B\} := AB + BA$.

In addition, we use calligraphic fonts, such as $\mathcal{L}$, to denote *superoperators*, which is also referred to as *quantum maps*. Superoperators are linear maps that take matrices to matrices. The *induced trace norm* of a superoperator $\mathcal{M}$, denoted by $\|\mathcal{M}\|_1$, is defined as

$$\|\mathcal{M}\|_1 := \max\{\|\mathcal{M}(A)\|_1 : \|A\|_1 \le 1\}. \tag{6}$$

The *diamond norm* of a superoperator $\mathcal{M}$, denoted by $\|\mathcal{M}\|_\diamond$, is defined as

$$\|\mathcal{M}\|_\diamond \coloneqq \|\mathcal{M} \otimes \mathcal{I}\|_1, \tag{7}$$

where $\mathcal{I}$ acts on the space with the same size as the space $\mathcal{M}$ acts on.

We denote by $\mathrm{C}^2[0,T]$ the class of twice continuously differential functions in $[0,T]$.

2.2 Algorithmic tools

2.2.1 Block-encoding and implementing completely-positive maps

Although we assume that the input of the operators of the Lindbladian are given by sparse-access oracles, it is convenient to use a more general input model when presenting the simulation algorithm. For a matrix $A \in \mathbb{C}^{2^n \times 2^n}$, we say that a unitary, denoted by U_A, is an (α, b, ϵ)-block-encoding of A if $\|A - \alpha(\langle 0|^{\otimes b} \otimes I)U_A(|0\rangle^{\otimes b} \otimes I)\| \leq \epsilon$, where the identity operator I is acting on n qubits. Intuitively, this unitary U_A is acting on $(n+b)$ qubits and A appears in the upper-left block of it, i.e., $U_A = \begin{pmatrix} A/\alpha & \cdot \\ \cdot & \cdot \end{pmatrix}$. Here, we refer to α as the *normalizing factor*.

Our simulation algorithm relies on the following technical tool from [37] for implementing completely positive maps given the block-encodings of its Kraus operators, which generalizes a similar tool in [15] where the Kraus operators are given as linear combinations of unitaries.

▶ **Lemma 2** (Implementing completely positive maps via block-encodings of Kraus operators [37]). *Let $A_1, \ldots, A_m \in \mathbb{C}^{2^n}$ be the Kraus operators of a completely positive map. Let $U_1, \ldots, U_m \in \mathbb{C}^{2^{n+n'}}$ be their corresponding (s_j, n', ϵ)-block-encodings, i.e.,*

$$\|A_j - s_j(\langle 0| \otimes I)U_j\,|0\rangle \otimes I)\| \leq \epsilon, \qquad \text{for all } 1 \leq j \leq m. \tag{8}$$

Let $|\mu\rangle \coloneqq \frac{1}{\sqrt{\sum_{j=1}^m s_j^2}} \sum_{j=1}^m s_j\,|j\rangle$. Then $\left(\sum_{j=1}^m |j\rangle\langle j| \otimes U_j\right)|\mu\rangle\,|0\rangle \otimes I$ implements this completely positive map in the sense that

$$\left\| I \otimes \langle 0| \otimes I \left(\sum_{j=1}^m |j\rangle\langle j| \otimes U_j\right)|\mu\rangle\,|0\rangle\,|\psi\rangle - \frac{1}{\sqrt{\sum_{j=1}^m s_j^2}} \sum_{j=1}^m |j\rangle\,A_j\,|\psi\rangle \right\| \leq \frac{m\epsilon}{\sqrt{\sum_{j=1}^m s_j^2}} \tag{9}$$

for all $|\psi\rangle$.

We also need the following lemma from [37] for obtaining a block-encoding of a linear combination of block-encodings.

▶ **Lemma 3** (Block-encoding of a sum of block-encodings [37]). *Suppose $A \coloneqq \sum_{j=1}^m y_j A_j \in \mathbb{C}^{2^n \times 2^n}$, where $A_j \in \mathbb{C}^{2^n \times 2^n}$ and $y_j > 0$ for all $j \in \{1, \ldots m\}$. Let U_j be an (α_j, a, ϵ)-block-encoding of A_j, and B be a unitary acting on b qubits (with $m \leq 2^b - 1$) such that $B\,|0\rangle = \sum_{j=0}^{2^b-1} \sqrt{\alpha_j y_j / s}\,|j\rangle$, where $s = \sum_{j=1}^m y_j \alpha_j$. Then a $(\sum_j y_j \alpha_j, a+b, \sum_j y_j \alpha_j \epsilon)$-block-encoding of $\sum_{j=1}^m y_j A_j$ can be implemented with a single use of $\sum_{j=0}^{m-1} |j\rangle\langle j| \otimes U_j + ((I - \sum_{j=0}^{m-1} |j\rangle\langle j|) \otimes I_{\mathbb{C}^{2^a}} \otimes I_{\mathbb{C}^{2^n}})$ plus twice the cost for implementing B.*

2.2.2 Optimization

For the current quantum-classical hybrid algorithm, we will couple a Lindblad simulation with a classical optimization algorithm. For this purpose, we work with the PAGD algorithm [27], which is based on Nesterov's accelerated gradient descent idea [44],

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k - \eta \nabla f(\boldsymbol{u}_k) + (1-\theta)\boldsymbol{v}_k, \quad \boldsymbol{v}_{k+1} \quad = \boldsymbol{u}_{k+1} - \boldsymbol{u}_k. \tag{10}$$

Here $\boldsymbol{u}_k$ is the kth iterate of the control variable. The idea in PAGD is to introduce a perturbation to the iterate when $\|\nabla f\| > \epsilon$ for some iterations, along with a negative curvature exploitation step.

There are two common goals for solving (nonconvex) optimization problems:

- $\boldsymbol{x}$ is called an ϵ-approximate first-order stationary point if $\|\nabla f(\boldsymbol{x})\| \leq \epsilon$.
- $\boldsymbol{x}$ is called an ϵ-approximate second-order stationary point if $\|\nabla f(\boldsymbol{x})\| \leq \epsilon$, $\lambda_{\min}(\nabla^2 f(\boldsymbol{x})) \geq -\sqrt{\varrho\epsilon}$. Here f is a ϱ-Hessian-Lipschitz function, i.e., $\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\| \leq \varrho\|\boldsymbol{x}-\boldsymbol{y}\|$ for any $\boldsymbol{x}$ and $\boldsymbol{y}$.

2.2.3 Quantum gradient estimation

With copies of $\rho(T)$, which will be obtained from Lindblad simulation algorithms, and sparse access to $\mathcal{O}$, we can obtain an estimated gradient value of $\widetilde{J}_1(\boldsymbol{u})$. The high-level strategy is to construct a probability oracle first, then construct a phase oracle with the probability oracle, and finally obtain the gradient by the phase oracle. The probability oracle and the phase oracle are defined as follows.

The Lindblad simulation algorithm leads to a purification of $\rho(T)$, i.e., $\rho(T) = \mathrm{tr}(|\rho_T\rangle\langle\rho_T|)$. It is clear that the regularization term in (2) is easy to compute. With the purification, we can express the first term as,

$$\widetilde{J}_1(\boldsymbol{u}) := \langle\rho_T|\,\mathcal{O}\otimes I\,|\rho_T\rangle\,. \tag{11}$$

Suppose $U_{\mathcal{O}}$ denotes the block encoding of $\mathcal{O}$, i.e. $\langle 0\,|\langle\psi_N\,|U_{\mathcal{O}}|\,0\rangle|\,\psi_N\rangle = \langle\psi_N|\mathcal{O}|\psi_N\rangle$. Let c-$U_{\mathcal{O}}$ be the controlled $U_{\mathcal{O}}$. Applying Hadamard test circuit $(H\otimes I)\,(\text{c}-U_{\mathcal{O}})\,(H\otimes I)$ acting on $|\rho_T\rangle$ produces

$$\sqrt{f(\boldsymbol{u})}\,|1\rangle\,|\phi_1(u)\rangle + \sqrt{1-f(\boldsymbol{u})}\,|0\rangle\,|\phi_0(\boldsymbol{u})\rangle \tag{12}$$

where $f(\boldsymbol{u}) := -\frac{1}{2}\,\langle\rho_T|\mathcal{O}|\rho_T\rangle + \frac{1}{2} = -\frac{1}{2}\widetilde{J}_1(\boldsymbol{u}) + \frac{1}{2}$. By Lemma 48 of [22], we can efficiently construct a block encoding of $\mathcal{O}$ with sparse access to $\mathcal{O}$. The 1/2 factor does not matter because the gradient will only be multiplied by a constant factor.

▶ **Definition 4** (Probability oracle). *Consider a function $f:\mathbb{R}^d\to[0,1]$. The probability oracle for f, denoted by U_f, is a unitary defined as*

$$U_f|\boldsymbol{x}\rangle|\mathbf{0}\rangle = |\boldsymbol{x}\rangle\left(\sqrt{f(\boldsymbol{x})}|1\rangle\,|\phi_1(\boldsymbol{x})\rangle + \sqrt{1-f(\boldsymbol{x})}|0\rangle\,|\phi_0(\boldsymbol{x})\rangle\right),$$

where $|\phi_1(\boldsymbol{x})\rangle$ and $|\phi_0(\boldsymbol{x})\rangle$ are arbitrary states.

▶ **Definition 5** (Phase oracle). *Consider a function $f\colon\mathbb{R}^d\to\mathbb{R}$. The phase oracle for f, denoted by $\mathcal{O}_f$, is a unitary defined as*

$$\mathcal{O}_f|\boldsymbol{x}\rangle|\mathbf{0}\rangle = e^{if(\boldsymbol{x})}|\boldsymbol{x}\rangle|\mathbf{0}\rangle$$

▶ **Theorem 6** (Constructing phase oracle with probability oracle, Theorem 14 of [21]). *Consider a function $f:\mathbb{R}^d\to[0,1]$. Let U_f be the probability oracle for f. Then, for any $\epsilon\in(0,1/3)$, we can implement an ϵ-approximate of the phase oracle $\mathcal{O}_f$ for f, denoted by $\widetilde{\mathcal{O}}_f$, such that $\|\widetilde{\mathcal{O}}_f|\psi\rangle|\boldsymbol{x}\rangle - \mathcal{O}_f|\psi\rangle|\boldsymbol{x}\rangle\| \leq \epsilon$, for all state $|\psi\rangle$. This implementation uses $O(\log(1/\epsilon))$ invocations to U_f and $U_f^\dagger$, and $O(\log\log(1/\epsilon))$ additional qubits.*

In order to interface the Lindblad simulation algorithm with a classical optimization method, one needs to estimate the gradient of the objective function. Similar to the approach in [35], we first represent the control variable as a piecewise linear function in time

$u_\beta(t) \approx \sum_{j=1}^N u_j B_j(t)$ with $B_j(t)$ being the standard shape function and u_j being a nodal function. The total number of steps N is proportional to the time duration T. We will use the improved Jordan's algorithm [28] using high order finite difference formulas [21]. Basically, the gradient estimation in [21] produces an estimate $\boldsymbol{g}(\boldsymbol{u})$, such that, $\|\nabla J_1(\boldsymbol{u}) - \boldsymbol{g}(\boldsymbol{u})\| < \epsilon$ with complexity $O(d/\epsilon)$, which is clearly better than a direct sampling approach. However, to achieve this complexity, the objective function needs to satisfy a derivative bound. Toward this end, we first establish an a priori bound for the derivative.

▶ **Lemma 7.** *Let $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_k) \in [N+1]^k$ be an index sequence*[3]*. The derivatives of the control function $\widetilde{J}_1$ with respect to the control variables satisfy:*

$$\left\| \frac{\partial^{\boldsymbol{\alpha}} \widetilde{J}_1}{\partial u_{\alpha_1} u_{\alpha_2} \cdots u_{\alpha_k}} \right\| \leq (k+1)! \left(\delta t \|\mu\|\right)^k . \tag{13}$$

This smoothness provides a basis for estimating the complexity of Jordan's algorithm [21],

▶ **Lemma 8** (Rephrased from Theorem 23 of [21])**.** *Suppose the access to $f\colon [-1,1]^N \to \mathbb{R}$ is given via a phase oracle O_f. If f is $(2m+1)$-times differentiable and for all $\boldsymbol{x} \in [-1,1]^N$, and $|\partial_{\boldsymbol{r}}^{2m+1} f(\boldsymbol{x})| \leq B$ for $\boldsymbol{r} = \boldsymbol{x}/\|\boldsymbol{x}\|$, then there exists a quantum algorithm that outputs an approximate gradient $\boldsymbol{g}$ such that $\|\boldsymbol{g} - \nabla f(\boldsymbol{0})\|_\infty \leq \epsilon$ with probability at least $1-\rho$ using*

$$\widetilde{O}\left(\max\left\{ \frac{N^{1/2} B^{1/(2m)} N^{1/(4m)} \log(N/\rho)}{\epsilon^{1+1/(2m)}}, \frac{m}{\epsilon} \right\} \right) \tag{14}$$

queries to O_f, and $\widetilde{O}(N)$ additional 1- and 2-qubit gates.

In particular, when $f(\boldsymbol{x})$ is a polynomial of degree no greater than $2m$, the query complexity to O_f becomes, $\widetilde{O}\left(\frac{m}{\epsilon}\right)$.

After adapting this algorithm to the objective function in Eq. (11), we find that,

▶ **Lemma 9.** *Let $\widetilde{J}_1$ be defined as in Eq.* (11)*. Suppose we are given access to the phase oracle $\mathcal{O}_{\widetilde{J}_1}$ for $\widetilde{J}_1$. Then, there exists a quantum algorithm that outputs an approximate gradient $\boldsymbol{g}$ such that $\|\boldsymbol{g} - \nabla \widetilde{J}_1\| \leq \epsilon_g$ with probability at least $1-\gamma$ using $\widetilde{\mathcal{O}}\left(n_{\mathrm{c}} T \log(N/\gamma)/\epsilon_g\right)$ queries to $\mathcal{O}_{\widetilde{J}_1}$, and $\widetilde{O}(N)$ additional 1- and 2-qubit gates.*

Proof. Although the derivative bound in Lemma 7 does not fulfill the condition in [21], we can apply Theorem 23 in [21]. By choosing the optimal value m, we arrive at the complexity bound. ◀

With the gradient estimated, we can now move to the optimization algorithm. The PAGD algorithm in [27] assumes the gradient- and Hessian-Lipschitz condition, which we will prove here for the control problem. In particular, the smoothness constant ℓ and the Hessian-Lipschitz constant ϱ can be approximated by the same technique as Lemma 7.

▶ **Lemma 10.** *Let $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_k) \in [N+1]^k$ be an index sequence, then $\tilde{J}_1$ is l-smooth and ρ-Hessian Lipschitz continuous, i.e.*

$$\|\boldsymbol{\nabla} \tilde{J}_1(\boldsymbol{u}) - \boldsymbol{\nabla} \tilde{J}_1(\boldsymbol{v})\| \leq l \|\boldsymbol{u} - \boldsymbol{v}\|, \textit{and} \quad \|\boldsymbol{\nabla}^2 \tilde{J}_1(\boldsymbol{u}) - \boldsymbol{\nabla}^2 \tilde{J}_1(\boldsymbol{v})\| \leq \varrho \|\boldsymbol{u} - \boldsymbol{v}\|. \tag{15}$$

The smoothness parameters are given by, $l = 3!(N+1)\delta t^2 \|\mu\|^2 \|\mathcal{O}\|, \quad \varrho = 4!(N+1)\delta t^3 \|\mu\|^3 \|\mathcal{O}\|$.

We refer the readers to the full version of this paper [26, Appendix B] for the proof of this lemma.

[3] For a precise definition of an index sequence, see Definition 4 of [21].

3 Simulating open quantum systems with time-dependent Lindbladian

Ref. [36, Section 6] sketched a method for simulating open quantum systems with time-dependent Lindbladian. In this section, we present the details of this simulation algorithm.

Motivated by the time scaling idea in [5], we define a change-of-variable function as

$$\mathrm{var}(t) \coloneqq \int_0^t \mathrm{d}s \, \|\mathcal{L}(s)\|_{\mathrm{be}}. \tag{16}$$

By simulating the Lindblad dynamics on the new time scale, the overall complexity exhibits a better dependence on the norm of the Lindbladians in time. To this end, we need the following oracle to perform the inverse change-of-variable:

$$\mathcal{O}_{\mathrm{var}} \ket{t}\ket{z} = \ket{t}\ket{z \oplus \mathrm{var}^{-1}(t)}. \tag{17}$$

In addition, we need the following oracle to obtain the normalizing constant $\alpha_0(t)$ for $H(t)$ and $\alpha_j(t)$ for $L_j(t)$: for all $j = [m]$,

$$\mathcal{O}_{H,\mathrm{norm}} \ket{t}\ket{z} = \ket{t}\ket{z \oplus \alpha_0(t)}, \quad \text{and} \quad \mathcal{O}_{L_j,\mathrm{norm}} \ket{t}\ket{z} = \ket{t}\ket{z \oplus \alpha_j(t)}. \tag{18}$$

As in [36], we define the *block-encoding norm* for a Lindbladian $\mathcal{L}$, denoted by $\|\mathcal{L}\|_{\mathrm{be}}$ for normalization purposed:

$$\|\mathcal{L}\|_{\mathrm{be}} \coloneqq \alpha_0 + \frac{1}{2}\sum_{j=1}^{m} \alpha_j^2. \tag{19}$$

The goal of this section is to prove the following theorem.

▶ **Theorem 11.** *Suppose we are given an $(\alpha_0(t), a, \epsilon')$-block-encoding $U_{H(t)}$ of $H(t)$, and an $(\alpha_j(t), a, \epsilon')$-block-encoding $U_{L_j(t)}$ for each $L_j(t)$ for all $0 \le t \le T$. Let $\|\mathcal{L}\|_{\mathrm{be},1}$ be defined as $\|\mathcal{L}\|_{\mathrm{be},1} \coloneqq \int_0^T \mathrm{d}\tau \, \|\mathcal{L}(\tau)\|_{\mathrm{be}}$, Suppose further that $\epsilon' \le \epsilon/(2t(m+1))$. Then, there exists a quantum algorithm that outputs a purification of $\tilde{\rho}_T$ of $\tilde{\rho}(T)$ where $\left\|\tilde{\rho}(T) - \mathcal{T}e^{\int_0^T \mathrm{d}\tau\, \mathcal{L}(\tau)}(\rho_0)\right\|_1 \le \epsilon$ using*

$$O\left(\|\mathcal{L}\|_{\mathrm{be},1}\left(\frac{\log\left(\|\mathcal{L}\|_{\mathrm{be},1}/\epsilon\right)}{\log\log\left(\|\mathcal{L}\|_{\mathrm{be},1}/\epsilon\right)}\right)^2\right) \tag{20}$$

queries to $U_{H(t)}$, $U_{L_j(t)}$, $\mathcal{O}_{\mathrm{var}}$, $\mathcal{O}_{H,\mathrm{norm}}$, and $\mathcal{O}_{L_j,\mathrm{norm}}$, and $\widetilde{O}\left((m+n)\|\mathcal{L}\|_{\mathrm{be},1}\right)$ additional 1- and 2-qubit gates, where n is the number of qubits the Lindbladian is acting on.

3.1 High-level overview of the simulation algorithm

Here we briefly outline the techniques that led to the stated complexity. Let the Hamiltonian $H(t) = H_0 + \sum_\beta u_\beta(t)\mu_\beta$, we rewrite equation (1) as follows

$$\frac{d}{dt}\rho = \mathcal{L}(t)(\rho) \coloneqq -i[H(t),\rho] + \sum_{j=1}^{m}(L_j(t)\rho L_j^\dagger(t) - \frac{1}{2}\{L_j(t)^\dagger L_j(t), \rho\}) \tag{21}$$

$$= \mathcal{L}_D(t)(\rho) + \mathcal{L}_J(t)(\rho). \tag{22}$$

Here we have decomposed the Lindbladian into a drift term $\mathcal{L}_{\mathrm{D}}(t)$ and a jump term $\mathcal{L}_{\mathrm{J}}(t)$:

$$\mathcal{L}_{\mathrm{D}}(t)(\rho) = -i[H(t),\rho] - \frac{1}{2}\sum_{j=1}^{m}\{L_j(t)^\dagger L_j(t),\rho\} = J(t)\rho + \rho J(t)^\dagger, \tag{23}$$

$$\mathcal{L}_{\mathrm{J}}(t)(\rho) = \sum_{j=1}^{m} L_j(t)\rho L_j(t)^\dagger, \tag{24}$$

where $J(t) \coloneqq -iH(t) - \frac{1}{2}\sum_{j=1}^{m} L_j(t)^\dagger L_j(t)$.

With the known initial value $\rho(0) = \rho_0$, the solution of Eq. (22) can be written as the linear combination of the following equations.

$$\begin{cases}\partial_t\rho &= \mathcal{L}_{\mathrm{D}}(t)(\rho)\\ \rho(0) &= \rho_0\end{cases}, \quad \text{and} \quad \begin{cases}\partial_t\rho &= \mathcal{L}_{\mathrm{D}}(t)(\rho) + \mathcal{L}_{\mathrm{J}}(t)(\rho)\\ \rho(0) &= 0\end{cases}. \tag{25}$$

Specifically, for the first part of Eq. (25), the density operator follows $\rho(t) = V(0,t)\rho_0 V(0,t)^\dagger = \mathcal{K}[V(0,t)](\rho_0)$, where $V(s,t) = \mathcal{T}e^{\int_s^t J(\tau)\mathrm{d}\tau}$ is the time-ordered exponential of J. A brief introduction of time-ordered exponential can be found in in the full version of this paper ([26, Appendix A]). For the second part of Eq. (25), the density operator follows $\rho(t) = \int_0^t g(t,s)\mathrm{d}s$, where the function $g(t,s)$ satisfying

$$\partial_t g(t,s) = \mathcal{L}_{\mathrm{D}}(t)(g(t,s)), \text{ and } \lim_{t\to s} g(t,s) = \mathcal{L}_{\mathrm{J}}(s)(\rho(s)). \tag{26}$$

By using time-ordered evolution operator and Duhamel's principle, the solution of Eq. (21) can be expressed as

$$\rho(t) = \mathcal{K}[V(0,t)](\rho_0) + \int_0^t \mathcal{K}[V(s,t)](\mathcal{L}_{\mathrm{J}}(s)(\rho(s)))\,\mathrm{d}s. \tag{27}$$

The time-ordered exponential $V(0,t)$ can be approximated by the truncated Dyson series (see the full version of this paper [26, Appendix A.1] for details),

$$V(0,t) = \mathcal{T}e^{\int_0^t J(\tau)\mathrm{d}\tau} \approx \sum_{k=0}^{K}\frac{1}{k!}\mathcal{T}\int_0^t \mathrm{d}\boldsymbol{\tau} J(\tau_k)\cdots J(\tau_1), \tag{28}$$

where $\mathcal{T}\int_0^t \mathrm{d}\boldsymbol{\tau}(\cdot)$ denote an integration over a k-tuple of time variables $(\tau_1,\ldots,\tau_k)$ while keeping the time ordering: $\tau_1 \leq \tau_2 \leq \cdots \leq \tau_k$. Thus,

$$V(s,t) = \mathcal{T}e^{\int_s^t J(\tau)\mathrm{d}\tau} = \mathcal{T}e^{\int_0^{t-s} J(s+\tau)\mathrm{d}\tau} \tag{29}$$

$$\approx \sum_{k=0}^{K}\frac{1}{k!}\int_0^{t-s}\mathrm{d}\boldsymbol{\tau}\mathcal{T}[J(\tau_k)\cdots J(\tau_1)]. \tag{30}$$

As in [30], we use the rectangle rule to approximate the integral in the truncated Dyson series. Note that more efficient quadratures could be potentially used as we use later approximation the integral in Eq. (27), for instance, the scaled Gaussian quadrature; however such methods require upper bounds on higher-order derivatives of $J(t)$, which are not readily available.

By applying Duhamel's principle (see Eq. (27)) several times, we obtain the following approximation with notations introduced in [10].

$$\mathcal{G}_K(t) \coloneqq \mathcal{K}[V(0,t)] + \sum_{k=1}^{K}\int_{0\leq s_1\leq\cdots\leq s_k\leq t}\mathcal{F}_k(s_k,\ldots,s_1)\,\mathrm{d}s_1\cdots\mathrm{d}s_k, \tag{31}$$

where

$$\mathcal{F}_k(s_k,\dots,s_1) := \mathcal{K}[V(s_k,t)]\mathcal{L}_{\mathrm{J}}(s_k)\cdots\mathcal{K}[V(s_1,s_2)]\mathcal{L}_{\mathrm{J}}(s_1)\mathcal{K}[V(0,s_1)]. \tag{32}$$

This yields an approximation of the evolution superoperator $\rho(t) \approx \mathcal{G}_K(t)(\rho(0))$. The key observation is that $\mathcal{F}_k$ is a composition of CPTP maps. The second term of $\mathcal{G}_K(t)$ can be approximated by using truncated Dyson series.

3.2 Detailed constructions

In this subsection, we present the construction of the time-dependent Lindbladian simulation algorithm. For the sake of conciseness, we omit the convoluted details in treating the time-ordering of the truncated Dyson series. All these details can be found in the full version of this paper [26, Appendix D].

The scaled evolution time

Recall that we introduced the rescaled time in Eq. (16), and let define $\hat{t}$ as

$$\hat{t} = \mathrm{var}(t) = \int_0^t \mathrm{d}s\, \|\mathcal{L}(s)\|_{\mathrm{be}}. \tag{33}$$

Correspondingly, we follow the rescaled Lindblad equation, by defining $\hat{\rho}(\hat{t}) = \rho((\mathrm{var}^{-1}(\hat{t}))$, which, from Eq. (1), satisfies the equation

$$\frac{\mathrm{d}}{\mathrm{d}\hat{t}}\hat{\rho}(\hat{t}) = \widehat{\mathcal{L}}(\hat{t})\hat{\rho}(\hat{t}), \tag{34}$$

where the rescaled Lindbladian is as,

$$\widehat{\mathcal{L}}(\hat{t}) = \frac{\mathcal{L}(\mathrm{var}^{-1}(\hat{t}))}{\left\|\mathcal{L}(\mathrm{var}^{-1}(\hat{t}))\right\|_{\mathrm{be}}}. \tag{35}$$

This rescaling can be achieved by defining

$$\widehat{H}(\hat{t}) := \frac{H(\mathrm{var}^{-1}(\hat{t}))}{\left\|\mathcal{L}(\mathrm{var}^{-1}(\hat{t}))\right\|_{\mathrm{be}}}, \text{ and } \widehat{L}(\hat{t}) := \frac{L(\mathrm{var}^{-1}(\hat{t}))}{\sqrt{\left\|\mathcal{L}(\mathrm{var}^{-1}(\hat{t}))\right\|_{\mathrm{be}}}}. \tag{36}$$

The *scaled effective Hamiltonian* (not Hermitian), denoted by $\widehat{J}(\hat{t})$, is therefore defined as

$$\widehat{J}(\hat{t}) := \frac{J(\mathrm{var}^{-1}(\hat{t}))}{\left\|\mathcal{L}(\mathrm{var}^{-1}(\hat{t}))\right\|_{\mathrm{be}}}. \tag{37}$$

As a result, simulating $\widehat{\mathcal{L}}$ for time $\hat{t} = \mathrm{var}(t)$ is equivalent to simulating $\mathcal{L}$ for time t. Moreover, the block-encoding-norm of $\widehat{\mathcal{L}}$ is at most 1 because of Eq. (35).

To simplify the notation, in the remainder of this section we assume the Lindbladian is already scaled so that we can drop the $\widehat{\cdot}$ notation for the scaled operators and evolution time.

LCU construction

Let $U_J(t)$ be an (α, a, ϵ)-block-encoding of $J(t)$. Given the oracles as in Eqs. (4) and (5), the unitary $\sum_t |t\rangle\langle t| \otimes U_J(t)$ for discretized times t can be implemented. Using Lemma 3, a block-encoding of $V(s,t)$ can also be implemented. More specifically, we use the rectangle rule as in [30] to approximate the integrals in Eq. (30):

$$\widetilde{V}(s,t) = \sum_{k=0}^{K'} \frac{(t-s)^k}{M^k k!} \sum_{j_1,\dots,j_k=0}^{M-1} \mathcal{T} J(t_{j_k})\cdots J(t_{j_1}). \tag{38}$$

Here the time-ordered term is defined as follows, for each tuple $t_k, t_{k-1}, \ldots, t_1$,

$$\mathcal{T} J(t_k) \cdots J(t_1) = J(t_{j_k}) \cdots J(t_{j_1}),$$

where $t_{j_k}, \ldots, t_{j_1}$ is the permutation of $t_k, t_{k-1}, \ldots, t_1$ that is in ascending order.

The error of the above approximation is bounded by

$$\left\| V(s,t) - \widetilde{V}(s,t) \right\| \leq O\left(\frac{(t-s)^{K'+1}}{(K'+1)!} + \frac{(t-s)^2 \dot{J}_{\max}}{M} \right), \tag{39}$$

where $\dot{J}_{\max} := \max_{\tau \in [0,t]} \left\| \frac{\mathrm{d}J(\tau)}{\mathrm{d}\tau} \right\|$.

Now, we need to approximate the integrals in Eq. (31). In [36], Gaussian quadratures were used to approximate similar integrals in the time-independent case, which yields a simpler LCU construction. Unfortunately, using such efficient quadrature rules in the time-dependent case requires bounding the norm of high-order derivatives of $V(s,t)$, which is not directly given. Instead, we use the simple Riemann sums for treating the integrals, where the LCU constructions follow closely from the ones in [30].

More specifically, we uniformly divide the evolution time t into q intervals, and let $t_j = tj/q$ for $j \in \{0, \ldots, M-1\}$. Assuming $V(s,t)$ is implemented perfectly, we consider the following superoperator,

$$\frac{t^k}{k! q^k} \sum_{j_1, \ldots, j_k = 0}^{q} \mathcal{T} \mathcal{F}_k(t_{j_k}, \ldots, t_{j_1}), \tag{40}$$

which approximates the integrals in Eq. (31). To bound the quality of this approximation, we need to bound the derivative of $\mathcal{F}_k$. We begin by bounding $\|V(0,t)\|$, which can be deduced from the stability of the differential equation $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{y} = J(t)\boldsymbol{y}$, which can be studied by examining the eigenvalues of the Hermitian part of $J(t)$ [7, Lemma 1]. Since the Hermitian part of $J(t)$ is semi-negative definite, one has $\|\boldsymbol{y}(t)\| \leq \|\boldsymbol{y}(0)\|$, which implies that

$$\|V(0,t)\| \leq 1. \tag{41}$$

Since $\frac{\mathrm{d}}{\mathrm{d}t} V(0,t) = J(t) V(0,t)$, the derivative of $V(0,t)$ can be bounded by

$$\left\| \frac{\mathrm{d}}{\mathrm{d}t} V(0,t) \right\| \leq \dot{J}_{\max}. \tag{42}$$

We further consider $\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}[V(0,t)]$. For any operator A with $\|A\|_1 = 1$, we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}[V(0,t)](A) = \left(\frac{\mathrm{d}}{\mathrm{d}t} V(0,t) \right) A V(0,t)^\dagger + V(0,t) A \frac{\mathrm{d}}{\mathrm{d}t} V(0,t)^\dagger. \tag{43}$$

We then have $\left\| \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}[V(0,t)](A) \right\|_1 \leq 2\dot{J}_{\max}$, which follows from Eqs. (41) and (42) and the fact that $\|BAC\|_1 \leq \|B\| \|A\|_1 \|C\|$ for matrices A, B, C. This bound easily extends to the diamond norm by tensoring the Kraus operator with an identity operator to extend it to a larger space. Hence, we have

$$\left\| \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}[V(0,t)](A) \right\|_\diamond \leq 2\dot{J}_{\max}. \tag{44}$$

Let $\dot{L}_{j,\max}$ be defined as $\dot{L}_{j,\max} := \max_{\tau \in [0,t]} \left\| \frac{\mathrm{d}}{\mathrm{d}\tau} L_j(\tau) \right\|$. Then, using similar arguments, we can bound the derivative of $\mathcal{L}_{\mathrm{J}}(t)$ as

$$\left\| \frac{\mathrm{d}}{\mathrm{d}t} \mathcal{L}_{\mathrm{J}}(t) \right\|_\diamond \leq 2 \sum_{j=1}^{m} \dot{L}_{j,\max}, \tag{45}$$

where we have assumed that the Lindbladian is scaled as in Eq. (35), i.e., $\|L_j\| \le 1$. For the derivative of $\mathcal{F}_k$, we have

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t_j}\mathcal{F}_k \\
&= \mathcal{K}[V(t_k,t)]\mathcal{L}_{\mathrm{J}}(t_k)\cdots\frac{\mathrm{d}}{\mathrm{d}t_j}\left(\mathcal{K}[V(t_j,t_{j+1})]\mathcal{L}_{\mathrm{J}}(t_j)\mathcal{K}[V(t_{j-1},t_j)]\right)\mathcal{L}_{\mathrm{J}}(t_{j-1})\cdots\mathcal{K}[V(0,t_1)] \\
&= \mathcal{K}[V(t_k,t)]\mathcal{L}_{\mathrm{J}}(t_k)\cdots\frac{\mathrm{d}}{\mathrm{d}t_j}(\mathcal{K}[V(t_j,t_{j+1})])\mathcal{L}_{\mathrm{J}}(t_j)\mathcal{K}[V(t_{j-1},t_j)]\mathcal{L}_{\mathrm{J}}(t_{j-1})\cdots\mathcal{K}[V(0,t_1)] \\
&+ \mathcal{K}[V(t_k,t)]\mathcal{L}_{\mathrm{J}}(t_k)\cdots\mathcal{K}[V(t_j,t_{j+1})]\frac{\mathrm{d}}{\mathrm{d}t_j}(\mathcal{L}_{\mathrm{J}}(t_j))\mathcal{K}[V(t_{j-1},t_j)]\mathcal{L}_{\mathrm{J}}(t_{j-1})\cdots\mathcal{K}[V(0,t_1)] \\
&+ \mathcal{K}[V(t_k,t)]\mathcal{L}_{\mathrm{J}}(t_k)\cdots\mathcal{K}[V(t_j,t_{j+1})]\mathcal{L}_{\mathrm{J}}(t_j)\frac{\mathrm{d}}{\mathrm{d}t_j}(\mathcal{K}[V(t_{j-1},t_j)])\mathcal{L}_{\mathrm{J}}(t_{j-1})\cdots\mathcal{K}[V(0,t_1)].
\end{aligned} \tag{46}$$

Again, assume the Lindbladian is scaled as in Eq. (35), the above expression of $\frac{\mathrm{d}}{\mathrm{d}t_j}\mathcal{F}_k$ together with Eqs. (44) and (45) implies that $\left\|\frac{\mathrm{d}}{\mathrm{d}t_j}\mathcal{F}_k\right\|_\diamond \le 4\dot{J}_{\max} + 2\sum_{j=1}^m \dot{L}_{j,\max}$. This implies that the error for using the Riemann sums can be bounded by

$$\begin{aligned}
&\left\|\mathcal{G}_K - \mathcal{K}[V(0,t)] - \frac{t^k}{k!q^k}\sum_{k=1}^K\sum_{j_1,\ldots,j_k=0}^q \mathcal{T}\mathcal{F}_k(t_{j_k},\ldots,t_{j_1})\right\|_\diamond \\
&= \left\|\sum_{k=1}^K\int_{0\le s_1\le\cdots\le s_k\le t}\mathcal{F}_k(s_k,\ldots,s_1)\,\mathrm{d}s_1\cdots\mathrm{d}s_k - \frac{t^k}{k!q^k}\sum_{k=1}^K\sum_{j_1,\ldots,j_k=0}^q \mathcal{T}\mathcal{F}_k(t_{j_k},\ldots,t_{j_1})\right\|_\diamond \\
&\le \sum_{k=1}^K\frac{t^2}{q}\cdot\left(4\dot{J}_{\max} + 2\sum_{j=1}^m \dot{L}_{j,\max}\right) && (47) \\
&= \frac{Kt^2}{q}\cdot\left(4\dot{J}_{\max} + 2\sum_{j=1}^m \dot{L}_{j,\max}\right). && (48)
\end{aligned}$$

In addition, it is easy to see that the error caused by using Duhamel's principle is

$$\left\|e^{\mathcal{T}\int_0^t \mathrm{d}\tau\,\mathcal{L}(\tau)} - \mathcal{G}_K\right\|_\diamond \le \frac{(2t)^{K+1}}{(K+1)!}. \tag{49}$$

It follows from Eqs. (48) and (49) that

$$\begin{aligned}
&\left\|e^{\mathcal{T}\int_0^t \mathrm{d}\tau\,\mathcal{L}(\tau)} - \mathcal{K}[V(0,t)] - \frac{t^k}{k!q^k}\sum_{j_1,\ldots,j_k=0}^q \mathcal{T}\mathcal{F}_k(t_{j_k},\ldots,t_{j_1})\right\|_\diamond \\
\le& \frac{(2t)^{K+1}}{(K+1)!} + \frac{Kt^2}{q}\left(4\dot{J}_{\max} + 2\sum_{j=1}^m \dot{L}_{j,\max}\right).
\end{aligned} \tag{50}$$

Finally, we have the following LCU form:

$$\widetilde{\mathcal{G}}_K \coloneqq \mathcal{K}[\widetilde{V}(0,t)] + \sum_{k=1}^K\frac{t^k}{q^k}\sum_{j_1,\ldots,j_k=0}^q \widetilde{\mathcal{F}}_k(t_{j_k},\ldots,t_{j_1}), \tag{51}$$

where $\widetilde{\mathcal{F}}_K$ is an approximation of Eq. (32) by using $\widetilde{V}(s,t)$ instead of $V(s,t)$, i.e.,

$$\widetilde{\mathcal{F}}_k(s_k,\ldots,s_1) := \mathcal{K}[\widetilde{V}(s_k,t)]\mathcal{L}_J(s_k)\mathcal{K}[\widetilde{V}(s_{k-1},s_k)]\cdots\mathcal{K}[\widetilde{V}(s_1,s_2)]\mathcal{L}_J(s_1)\mathcal{K}[\widetilde{V}(0,s_1)]. \quad (52)$$

We use the same compression scheme as in [30] to deal with the time-ordering in Eqs. (38) and (51). Note that implementing the LCU requires additional $O(KK'm(\log M + \log q + n))$ 1- and 2-qubit gates.

Complexity analysis

We first analyze the normalizing constant for the LCU implementation. Recall that we are working with scaled operators, so the normalizing factors are at most 1. For the implement of $V(0,\hat{t})$, we can use, for example, the LCU construction involving quantum sort as in [30] for implement Eq. (38). If we further assume the implementation uses an infinite Dyson series, the normalizing constants of the block-encoding $\mathcal{K}[V(0,t)]$ is upper bounded by

$$\sum_{k=0}^{\infty} \frac{t^k}{k!} = e^t. \quad (53)$$

As a result, the sum-of-squares of the normalizing constants of the Kraus operators of $\mathcal{F}_k(\hat{t}_k,\ldots,\hat{t}_1)$ can be bounded by

$$\sum_{j_1,\ldots,j_k=0}^{m} e^{2(t-s_k)}e^{2(s_k-s_{k-1})}\ldots e^{2(s_1-0)} = e^{2t}. \quad (54)$$

Recall that the normalizing constant for L_j is 1 since the Lindbladian is rescaled. For the second term in Eq. (31), the sum-of-squares of the normalizing constants of the Kraus operators can be bounded by

$$e^{2t}\frac{t^k}{k!q^k}q^k = e^{2t}\frac{t^k}{k!}. \quad (55)$$

By Eqs. (53) and (55), we have that the sum-of-squares of the normalizing constants of the Kraus operators of the LCU in Eq. (31) can then be bounded by $e^{2t} + \sum_{k=1}^{K} e^{2t}\frac{t^k}{k!} \leq e^{2t} + e^{3t}$.

Therefore, it suffices to set $t = \Omega(1)$ to achieve constant success probability when using Lemma 2. Then, we use the oblivious amplitude amplification for channels [15] to boost the success probability to 1 with constant applications of Lemma 2. For the error bound in Eq. (50), assume for now that the second error term is dominated by the first (by some choice of q to be determined). It suffices to set $K = \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}$ to make the total error at most $\epsilon/2$, because of the choice of $t = \Omega(1)$. The choice of q satisfies

$$q = \Theta\left(\frac{2K}{\epsilon}\left(4\dot{J}_{\max} + 2\sum_{j=1}^{m}\dot{L}_{j,\max}\right)\right). \quad (56)$$

Now, we deal with the error from truncated Dyson series and Riemann sum to implement $V(s,t)$. By Eq. (38), we can choose M large enough (determined later) so that the second error term is dominated by the first. Then, using [36, Lemma 7], we have

$$\left\|\mathcal{F}_k(s_k,\ldots,s_1) - \widetilde{\mathcal{F}}_k(s_k,\ldots,s_1)\right\|_{\diamond} \leq \frac{8e^t}{(K'+1)!}2^{k+1}t^{K'+1}. \quad (57)$$

Further, using the analysis as in [36], we can bound the total approximation error (with appropriate choice of M to be determined later) as

$$\left\|\mathcal{T}e^{\int_0^t \mathcal{L}(\tau)\mathrm{d}\tau} - \widetilde{\mathcal{G}}_K\right\|_\diamond \le \frac{32e^{5t}t^{K'+2}}{(K'+1)!}. \tag{58}$$

With the choice of $t = \Omega(1)$, we can choose $K' = \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}$ so that the total error is bounded by ϵ. For the choice of M, we need to make sure the second error term in Eq. (38) is dominated by the first term. Hence we can choose $M = \Theta\left(\frac{j_{\max}}{\epsilon}\right)$.

It remains to analyze the cost for the LCU implementation, which is the same as the analyses in [36] and [30]. Note that the dependence on M is logarithmic if the compressed scheme is used in [30] for implementing $\widetilde{V}(s,t)$. The total gate cost is now upper bounded by $O(KK'm(\log M + \log q + n))$. Further note that the error ϵ' brought by the block-encoding can be eventually transferred to $\mathcal{L}$ causing an $(m+1)\epsilon'$ error on $\mathcal{L}$ in terms of the diamond norm, and the accumulated error for evolution time t is then at most $t(m+1)\epsilon'$. As a result, choosing $\epsilon' \le \epsilon/(2t(m+1))$ suffices to ensure the total error is at most ϵ.

Recall that the above analysis is based on the scaled version of $\mathcal{L}$ defined in Eq. (35), and the evolution time is scaled as in Eq. (33). For arbitrary evolution time $\hat{t}$, we apply the above procedure $O(\hat{t})$ times with precision $\epsilon' = \epsilon/\hat{t}$. This gives the desired complexity in Theorem 11. Lastly, it is important to note that the LCU circuit yields a purification of $\rho(t)$. This completes the proof of Theorem 11.

Note that the above analysis easily extends to the simulation of the original Lindbladian without any scaling, where the complexity depends linearly on the product of evolution time and the maximum of the block-encoding norm of the Lindbladian. More specifically, we have the following corollary.

▶ **Corollary 12.** *Suppose we are given an $(\alpha_0(t), a, \epsilon')$-block-encoding $U_{H(t)}$ of $H(t)$, and an $(\alpha_j(t), a, \epsilon')$-block-encoding $U_{L_j(t)}$ for each $L_j(t)$ for all $t \ge 0$. Define $\|\mathcal{L}\|_{\mathrm{be},\infty}$ as $\|\mathcal{L}\|_{\mathrm{be},\infty} := \max_{\tau\in[0,T]} \|\mathcal{L}(\tau)\|_{\mathrm{be}}$. Suppose further that $\epsilon' \le \epsilon/(2T(m+1))$. There exists a quantum algorithm that outputs a purification of $\tilde{\rho}(T)$ where $\|\tilde{\rho}(T) - e^{\mathcal{T}\int_0^T \mathrm{d}\tau\, \mathcal{L}(\tau)}(\rho_0)\|_1 \le \epsilon$ using*

$$O\left(T\|\mathcal{L}\|_{\mathrm{be},\infty}\left(\frac{\log\left(T\|\mathcal{L}\|_{\mathrm{be},\infty}/\epsilon\right)}{\log\log\left(T\|\mathcal{L}\|_{\mathrm{be},\infty}/\epsilon\right)}\right)^2\right) \tag{59}$$

queries to $U_{H(t)}$, $U_{L_j(t)}$, $\mathcal{O}_{\mathrm{var}}$, $\mathcal{O}_{H,\mathrm{norm}}$, and $\mathcal{O}_{L_j,\mathrm{norm}}$, and $\widetilde{O}\left((m+n)T\|\mathcal{L}\|_{\mathrm{be},\infty}\right)$ additional 1- and 2-qubit gates. Here, n is the number of qubits the Lindbladian is acting on.

4 Simulations in the Interaction Picture

Many control problems involve a system Hamiltonian that contains a time-independent Hamiltonian that dominates the spectral norm $H(t)$, and thus the overall computational complexity. Motivated by the interaction picture approach for Hamiltonian simulations [41], we devise an approach to simulate the Lindblad dynamics. To formulate the problem, we assume that the Lindbladian admits the following decomposition:

$$\mathcal{L}(\cdot) = -i\left[H_1 + H_2(t), \cdot\right] + \sum_j L_j(\cdot)L_j^\dagger - \frac{1}{2}\left\{L_j^\dagger L_j, \cdot\right\}, \tag{60}$$

where H_1 is a time-independent free Hamiltonian, $H_2(t)$ is the coupling Hamiltonian which contains the control variables, and the dissipative terms still come from the interaction with the environment.

One such example is the control of an ion trap system [25], in which the model Hamiltonian consists of the following terms,

$$H_1 = \hbar \sum_{i=1}^{N} (\omega_{01}|1\rangle_i \langle 1| + \omega_{0e}|e\rangle_i \langle e|) + \hbar \sum_k \omega_k a_k^\dagger a_k \tag{61}$$

$$H_2(t) = \hbar\Omega_1 \cos(\boldsymbol{k}_1 \cdot \boldsymbol{r}_j - \omega_1 t - \varphi_1)(|0\rangle_j \langle e| + |e\rangle_j \langle 0|) \tag{62}$$

$$+ \hbar\Omega_2 \cos(\boldsymbol{k}_2 \cdot \boldsymbol{r}_j - \omega_2 t - \varphi_2)(|1\rangle_j \langle e| + |e\rangle_j \langle 1|), \tag{63}$$

and L_js includes $\lambda_{\text{heat}} a_j^\dagger$, $\lambda_{\text{damp}} a_j$ and $\lambda_{\text{dephase}} n_j$. The observation in [25] is that $\omega_{0e} \gg |\Omega_1|, |\Omega_2| \gg \lambda_{\text{heat}}, \lambda_{\text{damp}}, \lambda_{\text{dephase}}$.

Motivated by such applications, we assume that in Eq. (60),

$$\|H_1\| \gg \|H_2(t)\| \gg \|L_j\|. \tag{64}$$

In the interaction approach, e.g., [41], one simulates the density operator in the interaction picture, where the large magnitude of H_1 is absorbed into the slow Hamiltonian $H_2(t)$ and the jump operators. In this section, we provide detailed quantum algorithms for simulating the Lindbladian Eq. (60) in the interaction picture.

4.1 Lindbladian simulation in interaction picture

In light of Eq. (60), we first write the Lindbladian into two parts

$$\mathcal{L}(t) = \mathcal{L}_1 + \mathcal{L}_2(t) \tag{65}$$

where $\mathcal{L}_1$ contains a time-independent Hamiltonian term and $\mathcal{L}_2(t)$ can be a general Lindbladian term

$$\mathcal{L}_1(\cdot) = -i[H_1, \cdot] \tag{66}$$

$$\mathcal{L}_2(t)(\cdot) = -i[H_2(t), \cdot] + \sum_j L_j(\cdot)L_j^\dagger - \left\{L_j^\dagger L_j, \cdot\right\}. \tag{67}$$

Then the Lindblad master equation in Eq. (1) is equivalent to:

$$\frac{\mathrm{d}}{\mathrm{d}t} V_1\dagger(t_0,t)\rho V_1(t_0,t) = V_1^\dagger(t_0,t)\mathcal{L}_2(t)V_1(t_0,t)V_1^\dagger(t_0,t)\rho V_1(t_0,t), \tag{68}$$

where $V_1(t_0,t) = e^{-iH_1(t-t_0)}$, and $t \geq t_0$.

We can define $\rho_{\text{I}} = V_1^\dagger(t_0,t)\rho V_1(t_0,t)$ as the density operator in the interaction picture, and it satisfies the Lindblad equation, $\frac{d}{dt}\rho_{\text{I}}(t) = \mathcal{L}_{2,\text{I}}(t)\rho_{\text{I}}(t)$, where $\mathcal{L}_{2,\text{I}}(t) := V_1^\dagger(t_0,t)\mathcal{L}_2(t)V_1(t_0,t)$. Effectively, this transforms H_2 and $L_j(t)$ in Eq. (67) into an interaction picture as well.

By simulating the time evolution in the interaction picture and transforming it back to the original picture at last, we have

$$\rho(t) = \left(e^{\mathcal{L}_1(t-t_0)}\right)\left(\mathcal{T}e^{\int_{t_0}^{t} \mathcal{L}_{2,\text{I}}(s)\mathrm{d}s}\rho(t_0)\right), \tag{69}$$

where $\left(e^{\mathcal{L}_1(b-a)}\right)(\cdot) = V_1(a,b)(\cdot)V_1^{-1}(a,b)$. We can further decompose this evolution into N Trotter steps (with $\tau = (t-t_0)/N$),

$$\rho(t) = \prod_{i=0}^{N-1}\left(e^{\mathcal{L}_1\tau}\mathcal{T}e^{\int_{t_0+i\tau}^{t_0+(i+1)\tau}\mathcal{L}_{2,\mathrm{I}}(s)\mathrm{d}s}\right)\rho(t_0). \tag{70}$$

At a high level, Eq. (70) summarizes our simulation strategy in the interaction picture. The total time complexity is determined by the number of time steps N, and the time complexity in each step, which follows from our Lindbladian simulation algorithm in Section 3.

▶ **Theorem 13** (Modified from Corollary 12)**.** *Suppose we are given an (α_0, a, ϵ')-block-encoding U_H of H, and an (α_j, a, ϵ')-block-encoding U_{L_j} for each L_j. For all $\tau, \epsilon' \geq 0$ and $t\|\mathcal{L}(\tau)\|_{\mathrm{be},\infty} = \Theta(1)$, there exists a quantum algorithm for simulating $e^{\mathcal{L}\tau}$ using $O\left(\frac{\log\left(1/\epsilon'\right)}{\log\log(1/\epsilon')}\right)$ queries to U_H and U_{L_j} and $O\left(m\left(\frac{\log\left(1/\epsilon'\right)}{\log\log(1/\epsilon')}\right)^2\right)$ additional 1- and 2-qubit gates.*

▶ **Lemma 14** (Error accumulation)**.** *Given that $A_j = W_j$ and $B_j = \mathcal{T}\left[e^{\int_{t_{j-1}}^{t_j}\mathcal{L}_1(s)\mathrm{d}s}\right]$ are bounded $\|W_j\| \leq 1$, $\|B_j\| \leq 1$, and error in each segment is bounded by δ $\|A_j - B_j\| \leq \delta$. Then the accumulated error is*

$$\left\|\prod_{j}^{L} W_j - \mathcal{T}\left[e^{\int_0^t \mathcal{L}(s)\mathrm{d}s}\right]\right\| \leq L\delta. \tag{71}$$

Proof. The lemma holds by applying the triangle inequality

$$\left\|\prod_{j=1}^{L} A_j - \prod_{j=1}^{L} B_j\right\| \leq \sum_{k=1}^{L}\left(\prod_{j=1}^{k-1}\|A_j\|\right)\|A_k - B_k\|\left(\prod_{j=k+1}^{L}\|B_j\|\right). \tag{72}$$

◀

These results imply the following result for Lindbladian simulation in the interaction picture:

▶ **Theorem 15** (Query complexity of Lindbladian simulation in the interaction picture)**.** *Let $\mathcal{L}(t) = \mathcal{L}_1(t) + \mathcal{L}_2(t)$, with $\mathcal{L}_1(t)$ and $\mathcal{L}_2(t)$ defined by Eqs. (66) and (67) respectively. Assume the existence of a unitary oracle that implements the Hamiltonian and Lindbladian within the interaction picture, denoted U_{H^I} and $U_{L_j^I}$ which implicitly depends on the time-step size $\tau \in \mathcal{O}\left(||\mathcal{L}_2||_{\mathrm{be}}^{-1}\right)$ and number of quadrature points q, such that*

$$\left(\langle 0|_a \otimes \mathbf{1}_s\right) U_{H^I}\left(|0\rangle_a \otimes \mathbf{1}_s\right) = \sum_{j_k=1}^{q} |j_k\rangle\langle j_k| \otimes \frac{e^{iH_1\tau\hat{x}_{(j_k)}}H_2 e^{-iH_1\tau\hat{x}_{(j_k)}}}{\alpha_H} \tag{73}$$

$$\left(\langle 0|_a \otimes \mathbf{1}_s\right) U_{L_j^I}\left(|0\rangle_a \otimes \mathbf{1}_s\right) = \sum_{j_k=1}^{q} |j_k\rangle\langle j_k| \otimes \frac{e^{iH_1\tau\hat{x}_{(j_k)}}L_j e^{-iH_1\tau\hat{x}_{(j_k)}}}{\alpha_{L_j}}, \tag{74}$$

For $t \geq ||\mathcal{L}_2(t)||_{\mathrm{be}}\tau$, the time-evolution operator $\mathcal{T}e^{\int_0^t \mathcal{L}_1(s)+\mathcal{L}_2(s)\mathrm{d}s}$ may be approximated to error ϵ with the following cost.

1. *Simulations of $e^{-iH_1\tau}$: $\mathcal{O}\left(t||\mathcal{L}_2(t)||_{\mathrm{be},\infty}\right)$,*
2. *Queries to U_{H^I} and $U_{L_j^I}$: $\mathcal{O}\left(t||\mathcal{L}_2(t)||_{\mathrm{be},\infty}\frac{\log(t||\mathcal{L}_2(t)||_{\mathrm{be},\infty}/\epsilon)}{\log\log(t||\mathcal{L}_2(t)||_{\mathrm{be},\infty}/\epsilon)}\right)$,*
3. *Primitive gates: $\mathcal{O}\left(mt||\mathcal{L}_2(t)||_{\mathrm{be},\infty}\left(\frac{\log(t||\mathcal{L}_2(t)||_{\mathrm{be},\infty}/\epsilon)}{\log\log(t||\mathcal{L}_2(t)||_{\mathrm{be},\infty}/\epsilon)}\right)^2\right)$.*

Proof. Consider simulation strategy shown in Eq. (70), we uniformly divide the evolution time $[0,t]$ into $M = \lceil t\left\|\mathcal{L}_2(t)\right\|_{\text{be},\infty}\rceil$, time step $\tau = t/M$. Then $\tau\left\|\mathcal{L}_2(t)\right\|_{\text{be},\infty} = \Theta(1)$, which satisfies the pre-condition of Theorem 13. Therefore, using Theorem 13, the time and gate complexity of each time interval is $O\left(\frac{\log(1/\epsilon')}{\log\log(1/\epsilon')}\right)$ and $O\left(m\left(\frac{\log(1/\epsilon')}{\log\log(1/\epsilon')}\right)^2\right)$, respectively. Furthermore, by the error accmulation in Lemma 14, we choose $\epsilon' = \epsilon/t\left(\|\mathcal{L}\|_{\text{be}}\right)$ in order to bound the overall error by ϵ.

In addition, since we need to invoke $e^{-iH_1\tau}$ once every step, the invoking number equals to M and is hence bounded as claimed. ◀

4.2 Comparison of the simulation complexity with and without interaction picture

In this subsection, we compare the complexity with simulations of Lindblad dynamics with and without the interaction picture. For the Lindbladian decomposition shown in Eq. (65), suppose we have access to the oracles U_{H_1},$U_{H_2(t)}$, and U_{L_j}. According to Theorem 11, a direct simulation involves a time complexity

$$C_{\text{direct}} = O\left(t(C_1 + C_2)(\alpha_{L_1} + \alpha_{L_2})(\frac{\log\left(t(\alpha_{L_1} + \alpha_{L_2})/\epsilon\right)}{\log\log\left(t(\alpha_{L_1} + \alpha_{L_2})/\epsilon\right)})^2\right) \tag{75}$$

where $\alpha_1 = \left\|\mathcal{L}_1(t)\right\|_{\text{be},1}, \alpha_{L_2} = \left\|\mathcal{L}_2(t)\right\|_{\text{be},1}$; C_1 and C_2 representing the gate complexity of implement U_{H_1} and the maximum gate complexity of implement $U_{H_2(t)}, U_{L_j}$ respectively.

Meanwhile for the simulation algorithm in interaction picture, the time complexity is given by the following theorem.

▶ **Theorem 16** (Gate complexity of Lindbladian simulation in the interaction picture). *Suppose we are given U_{H_1},$U_{H_2(t)}$ and U_{L_j} block encoding of $H_1, H_2(t)$ and L_j respectively, such that e^{-iHs} is approximated to error ϵ using $C_{e^{-iH_1s}}[\epsilon] \in \mathcal{O}\left(|s|\log^{\gamma}(s/\epsilon)\right)$ gates for some $\gamma > 0$ and any $|s| \geq 0$.*

For all $t > 0$, the time-evolution Eq. (70) may be approximated to error ϵ with gate complexity

$$\begin{aligned} &C_{\text{interact}} \\ &= \mathcal{O}\left(\alpha_{L_2}t\left(C_2 + C_{e^{-iA/\alpha_{L_2}}}\left[\frac{\epsilon}{\alpha_{L_2}t\log\left(\alpha_{L_2}\right)}\right]\log\left(\frac{t\left(\alpha_{L_1} + \alpha_{L_2}\right)}{\epsilon}\right)\right)\frac{\log\left(\alpha_{L_2}t/\epsilon\right)}{\log\log\left(\alpha_{L_2}t/\epsilon\right)}\right) \\ &= \mathcal{O}\left(\alpha_{L_2}t\left(C_2 + C_{e^{-iA/\alpha_{L_2}}}[\epsilon]\right)\operatorname{polylog}\left(t\left(\alpha_{L_1} + \alpha_{L_2}\right)/\epsilon\right)\right) \end{aligned} \tag{76}$$

where $\alpha_1 = \left\|\mathcal{L}_1(t)\right\|_{\text{be},\infty}, \alpha_{L_2} = \left\|\mathcal{L}_2(t)\right\|_{\text{be},\infty} = \left\|\mathcal{L}_{2,I}(t)\right\|_{\text{be},\infty}$.

The proof follows by using $\alpha_{L_1}, \alpha_{L_{2,I}}$ to substitute α_A and α_B in Theorem 7 in [41], respectively.

We highlight that the assumption $C_{e^{-iH_1s}}[\epsilon] = O\left(|s|\log^{\gamma}(s/\epsilon)\right)$ imposes strong requirement on simulating the H_1 dynamics. With Hamiltonian simulation algorithm [5], gate complexity should be $C_{e^{-iH_1s}}[\epsilon] = \tilde{O}(||H_1||s)$. But here the assumption removes the $||H_1||$ dependence. This implies that the simulation of H_1 is supposed to be easy, the dynamics can be fast-forwarded. Nevertheless this assumption is valid in some common settings [41], for instance when H_1 is diagonal. Another assumption is Eq. (64), which implies that $\alpha_2 \ll \alpha_1$. By comparing Eq. (75) and Eq. (76) with this relation, we find the simulation strategy using the interacting picture has a better gate complexity. As long as these two assumptions hold, the simulation algorithm in the interaction picture can serve as an alternative to reduce the simulation complexity.

5 The Optimization Algorithm for Quantum Optimal Control

In this section, we present our main results for finding first- and second-order stationary points of the optimization problem induced by the quantum optimal control problem (2), which in general is nonconvex. We consider the accelerated gradient descent (AGD) method [27]. A key departure from a direct implement of AGD is that the gradient has to be estimated using the quantum algorithm [21], in which case, the gradient input is subject to noise. We believe that this result may be of general interest to the optimization community.

▶ **Theorem 17.** *Assume that the function $f(\cdot)$ is ℓ-smooth and ϱ-Hessian Lipschitz. There exists an absolute constant $c_{\max}$ such that for any $\delta > 0, \epsilon \leq \frac{\ell^2}{\varrho}, \Delta_f \geq f(\mathbf{x}_0) - f^\star$, if $\chi = \max\left\{1, \log\frac{d\ell\Delta_f}{\varrho\epsilon\delta}\right\}$, $c \geq c_{\max}$ and such that if we run modified PAGD ([26, Algorithm 2]) with the choice of parameters in [26, Appendix C.1] using an approximate gradient $\hat{\nabla} f(x)$ with error bounded at every step: $\|\nabla f(x) - \hat{\nabla} f(x)\| \leq \epsilon_g$ with*

$$\epsilon_g = \frac{\varrho^{1/8}}{\sqrt{2}\ell^{1/4}\chi^{3/2}c^{3/2}}\epsilon^{9/8}, \tag{77}$$

then with probability at least $1-\delta$, one of the iterates $\mathbf{x}_t$ will be an ϵ-first order stationary point in the following number of iterations:

$$O\left(\frac{\ell^{1/2}\varrho^{1/4}\left(f(\mathbf{x}_0) - f^*\right)}{\epsilon^{7/4}}\log^6\left(\frac{d\ell\Delta_f}{\varrho\epsilon\delta}\right)\right). \tag{78}$$

Furthermore, if the error bound of the gradient is chosen as, $\epsilon_g = \frac{\delta\chi^{-11}c^{-16}}{64\ell}\frac{\epsilon^3}{\sqrt{d}}\frac{1}{\Delta_f}$, then with probability at least $1-\delta$, one of the iterates $\mathbf{x}_t$ will be an ϵ-second order stationary point.

The proof of this theorem can be found in the full version of this paper [26, Appendix C.7]. Note that the complexity $\tilde{O}(1/\epsilon^{7/4})$ in [27] is the currently best-known result for finding first- and second-order stationary points using only gradient queries, and there is not much space to improve as [11] proved a lower bound $\Omega(1/\epsilon^{12/7})$ for deterministic algorithms with gradient queries when the function is gradient- and Hessian-Lipschitz. Our error bound $\tilde{O}(1/\epsilon^{9/8})$ in (77) is optimal (up to poly-logarithmic factors) for PAGD because up to a concentration inequality, it can give an algorithm for stochastic gradient descent with complexity $\tilde{O}(1/\epsilon^{7/4} \cdot (1/\epsilon^{9/8})^2) = \tilde{O}(1/\epsilon^4)$, which is optimal as there is a matching lower bound $\Omega(1/\epsilon^4)$ [3]. In other words, if the error $\epsilon^{9/8}$ can be further improved, it implies an algorithm for finding stationary points with better convergence than [27], the current state-of-the-art work on this.

The AGD algorithm relies on an estimate of the gradient. Toward this end, we first show that the objective function (11) from the quantum control problem is essentially a polynomial. The polynomial nature of the objective function allows us to use high-order finite difference methods to compute the gradient. In particular, a centered difference scheme with $2m+1$ points will produce an exact gradient for a polynomial of degree $2m$.

▶ **Lemma 18.** *Assume that the control function is expressed as a linear combination of shape function $b_j(t)$: $u(t) = \sum_{j=0}^N u_j b_j(t)$ and let $\boldsymbol{u} = (u_0, u_1, \ldots, u_N)$. Then the expectation in Eq. (11) from the Lindblad simulation algorithms from the previous section is a polynomial with degree $d = O\left(T \operatorname{polylog}\frac{1}{\epsilon}\right)$.*

Proof. We begin by examining the time-dependent unitary $V(0,t)$ in Duhamel's representation. Specifically, from Eq. (28), we see that the Dyson series approximation yields a polynomial of degree at most K. In addition, in the Kraus form approximation in Eq. (31), the operators $\mathcal{L}_{\mathrm{J}}(s)$ do not involve the control variable $\boldsymbol{u}$. Overall, the approximation $\mathcal{G}_K(t)$ in Eq. (31) constitutes a polynomial of degree at most K^2. Therefore, after applying $\mathcal{G}_K(\delta)$ for T/δ times to approximate the density operator at time T, we obtain a polynomial of degree at most $T\mathrm{polylog}\frac{1}{\epsilon}$. Here we have used the fact that $K = \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}$. Furthermore, when the gradient estimation algorithm in Lemma 8 is applied, the query complexity $\widetilde{O}\left(\frac{m}{\epsilon}\right)$ in Lemma 8 becomes $\widetilde{O}(T/\epsilon)$. ◀

6 Proof of Main Theorem

Finally, we outline the proof of our main theorem (Theorem 1). We first summarize our quantum algorithm as follows,

Algorithm 1 Quantum Algorithm for Open System Quantum Control.

```
Given $k_{\max}, \epsilon_g$ as in Theorem 17; set $u(t) = 0$
for t = 0,1,...,$k_{\max}$ do
    Use Theorem 11 and strategy in Section 2.2.3 to construct the phase oracle for $\widetilde{J}_1(\boldsymbol{u})$;
    Use Lemma 9 to estimate $\boldsymbol{g}^{(k)} \approx \nabla J\left(\boldsymbol{u}^{(k)}\right)$ with $||\boldsymbol{g}^{(k)} - J\left(\boldsymbol{u}^{(k)}\right)|| \leq \epsilon_g$;
    Update control variable with one step of modified PAGD ([26, Algorithm 2]);
end for
```

Now, we restated the main theorem and give its proof:

▶ **Theorem 19** (main theorem, restated)**.** *Assume there are n_{c} control functions $u_\beta(t) \in \mathrm{C}^2([0,T])$. Further assume*[4] *that $\|H_0\|, \|\mathcal{O}\|, \|\mu_\beta\|, \|L_j\| \leq 1$, and $\alpha \geq 2/T$. There exists a quantum algorithm that, with probability at least $2/3$*[5]*, solves problem (2) by:*

- *reaching a first-order stationary point $\|\nabla f\| < \epsilon$ with (1) using $\widetilde{O}\left(\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T}{\epsilon^{23/8}}\Delta_f\right)$ queries to $\mathcal{P}_{H_0}$ and $\mathcal{P}_{\mu_\beta}$, $\beta = 1, 2, \ldots, n_{\mathrm{c}}$, and $\widetilde{O}\left(mn\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T}{\epsilon^{23/8}}\Delta_f + n\frac{T^{3/2}}{\epsilon^{9/4}}\Delta_f\right)$ additional 1- and 2-qubit gates; or*
- *reaching a second-order stationary point using $\widetilde{O}\left(\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T^{7/4}}{\epsilon^5}\Delta_f\right)$ queries to $\mathcal{P}_{H_0}$ and $\mathcal{P}_{\mu_\beta}$, $\beta = 1, 2, \ldots, n_{\mathrm{c}}$ and $\widetilde{O}\left(mn\frac{n_{\mathrm{c}}\|\mathcal{L}\|_{be\ ,1}T^{7/4}}{\epsilon^5}\Delta_f + n\frac{T^{3/2}}{\epsilon^{9/4}}\Delta_f\right)$ additional 1- and 2-qubit gates.*

Here n_{c} and m are respectively the number of control variables and jump operators.

Proof. We denote gate complexity of control $U_{\mathcal{O}}$ by $C_{\mathrm{c}-U_{\mathcal{O}}}$, gate complexity of U_H, U_{L_j} by $C_{U_H, U_{L_j}}$, and gate complexity of quantum simulation by $C_{\varrho(t)}$. The gate complexity of preparing a state after Lindblad evolution is given by Theorem 11,

$$C_{\varrho(t)} = O\left(\|\mathcal{L}\|_{\mathrm{be},1}\frac{\log\left(\|\mathcal{L}\|_{\mathrm{be},1}/\epsilon\right)}{\log\log\left(\|\mathcal{L}\|_{\mathrm{be},1}/\epsilon\right)}\right)C_{U_H,U_{L_j}} + \tilde{O}\left(m\|\mathcal{L}\|_{\mathrm{be},1}n\right). \tag{79}$$

With copies of states $\varrho(t)$ and access to control $U_{\mathcal{O}}$ oracle, we can construct the gradient following Section 2.2.3. According to that section, we can construct the probability oracle with $\varrho(t)$, construct the phase oracle with probability oracle, and calculate the gradient with the phase oracle. The corresponding complexity is listed below:

[4] More generally, if $\|H_0\|, \|\mu\| = \Theta(\Lambda)$, it is equivalent to enlarge the time duration T by a factor $O(\Lambda)$.

[5] Using standard techniques, the success probability can be boosted to a constant arbitrarily close to 1 while only introducing a logarithmic factor in the complexity.

$$
\begin{aligned}
C_{U_{J_1}} &= C_{\varrho(t)} + O(1) + C_{c-U_O}, && (80)\\
C_{\mathcal{O}_{J_1}} &= O(\log 1/\epsilon) C_{U_{J_1}}, && (81)\\
C_{\nabla J} &= \tilde{O}(n_{\mathrm{c}} T \log(N/\gamma)/\epsilon) C_{\mathcal{O}_{J_1}} + \tilde{O}(N), && (82)
\end{aligned}
$$

where $1-\gamma$ is the successful probability of obtaining a gradient, n_{c} is the number of parameters, and N is the time steps $N = O\left(t^{3/2}/\epsilon^{1/2}\right)$ as in [35, Corollary 2.2]. Here we define $\gamma = \nu/k$, where ν is a small finite number and k is the iteration steps, which we will give below. Combining them together, we have

$$
\begin{aligned}
C_{\nabla J} = \tilde{O}\left(n_{\mathrm{c}} \frac{\|\mathcal{L}\|_{\mathrm{be},1} T \log \frac{N}{\gamma}}{\epsilon}\right) C_{U_H, U_{L_j}} + \tilde{O}(n_{\mathrm{c}} \frac{T \log \frac{N}{\gamma}}{\epsilon}) C_{\mathrm{c}-U_O} \\
+ \tilde{O}(m n n_{\mathrm{c}} \frac{\|\mathcal{L}\|_{\mathrm{be},1} T \log \frac{N}{\gamma}}{\epsilon} + N). \qquad (83)
\end{aligned}
$$

Here we reassign the gradient noise ϵ with ϵ_g to distinguish from the other errors.

$$
\begin{aligned}
C_{\nabla J} = \tilde{O}\left(n_{\mathrm{c}} \frac{\|\mathcal{L}\|_{\mathrm{be},1} T \log \frac{N}{\gamma}}{\epsilon_g}\right) C_{U_H, U_{L_j}} + \tilde{O}(n_{\mathrm{c}} \frac{T \log \frac{N}{\gamma}}{\epsilon_g}) C_{\mathrm{c}-U_O} \\
+ \tilde{O}(m n n_{\mathrm{c}} \frac{\|\mathcal{L}\|_{\mathrm{be},1} T \log \frac{N}{\gamma}}{\epsilon_g} + N). \qquad (84)
\end{aligned}
$$

With modified PAGD method ([26, Algorithm 2]), we can find a first or second order ϵ-stationary point within

$$
k = \tilde{O}\left(\frac{\ell^{1/2} \varrho^{1/4} \left(f\left(\mathbf{x}_0\right) - f^*\right)}{\epsilon^{7/4}}\right) \qquad (85)
$$

iterations by Theorem 17. For first ϵ-order stationary point, the gradient noise tolerance is $\epsilon_g = \frac{\varrho^{1/8}}{\sqrt{2}\ell^{1/4}\chi^{3/2}c^{3/2}} \epsilon^{9/8}$. For second order ϵ-order stationary point, it is $\epsilon_g = \frac{\delta \chi^{-11} c^{-16}}{64\ell} \frac{\epsilon^3}{\sqrt{d}} \frac{1}{\Delta_f}$.

In each iteration, we need to calculate ∇J once and calculate J once. Noticing that $C_J = O(C(\varrho(t)))$, we have

$$
C_{\mathrm{total}} = k \times (C_{\nabla J} + C_{\varrho(t)}). \qquad (86)
$$

Substitute Eqs. (79) and (84), (85) into Eq. (86) we finish the proof. Notice that in optimization, dimension $d = N$, and here we regard ℓ and ϱ as constants. ◀

References

1 Mohamed Abdelhafez, David I. Schuster, and Jens Koch. Gradient-based optimal control of open quantum systems using quantum trajectories and automatic differentiation. *Physical Review A*, 99(5):052327, 2019.

2 Claudio Altafini. Coherent control of open quantum dynamical systems. *Physical Review A*, 70(6):062321, 2004.

3 Yossi Arjevani, Yair Carmon, John C. Duchi, Dylan J. Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 199(1-2):165–214, 2023.

4 Leonardo Banchi and Gavin E Crooks. Measuring analytic gradients of general quantum evolution with the stochastic parameter shift rule. *Quantum*, 5:386, 2021.

5 Dominic W Berry, Andrew M Childs, Yuan Su, Xin Wang, and Nathan Wiebe. Time-dependent Hamiltonian simulation with L^1-norm scaling. *Quantum*, 4:254, 2020.

6 Samuel Boutin, Christian Kraglund Andersen, Jayameenakshi Venkatraman, Andrew J. Ferris, and Alexandre Blais. Resonator reset in circuit QED by optimal control for large open quantum systems. *Physical Review A*, 96(4):042315, 2017.

7 Fred Brauer. Perturbations of nonlinear systems of differential equations. *Journal of Mathematical Analysis and Applications*, 14(2):198–206, 1966.

8 Constantin Brif, Raj Chakrabarti, and Herschel Rabitz. Control of quantum phenomena: past, present and future. *New Journal of Physics*, 12(7):075008, 2010.

9 Tommaso Caneva, Tommaso Calarco, and Simone Montangero. Chopped random-basis quantum optimization. *Physical Review A*, 84(2):022326, 2011.

10 Yu Cao and Jianfeng Lu. Structure-preserving numerical schemes for Lindblad equations. *arXiv preprint*, 2021. `arXiv:2103.01194`.

11 Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points II: first-order methods. *Mathematical Programming*, 185(1-2):315–355, 2021.

12 Davide Castaldo, Marta Rosa, and Stefano Corni. Quantum optimal control with quantum computers: A hybrid algorithm featuring machine learning optimization. *Physical Review A*, 103(2):022613, 2021.

13 Raj Chakrabarti and Herschel Rabitz. Quantum control landscapes. *International Reviews in Physical Chemistry*, 26(4):671–735, 2007.

14 Andrew M Childs and Tongyang Li. Efficient simulation of sparse Markovian quantum dynamics. *Quantum Information & Computation*, 17(11-12):901–947, 2017.

15 Richard Cleve and Chunhao Wang. Efficient quantum algorithms for simulating Lindblad evolution. In *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2017.

16 Pierre De Fouquieres and Sophie G Schirmer. A closer look at quantum control landscapes and their implication for control optimization. *Infinite dimensional analysis, quantum probability and related topics*, 16(03):1350021, 2013.

17 Pierre de Fouquieres, Sophie G. Schirmer, Steffen J. Glaser, and Ilya Kuprov. Second order gradient ascent pulse engineering. *Journal of Magnetic Resonance*, 212(2):412–417, 2011.

18 Domenico d'Alessandro. *Introduction to quantum control and dynamics*. CRC press, 2021.

19 Xiaozhen Ge, Rebing Wu, and Herschel Rabitz. Optimization landscape of quantum control systems. *Complex System Modeling and Simulation*, 1(2):77–90, 2021.

20 D Geppert, L Seyfarth, and R de Vivie-Riedle. Laser control schemes for molecular switches. *Applied Physics B*, 79:987–992, 2004.

21 András Gilyén, Srinivasan Arunachalam, and Nathan Wiebe. Optimizing quantum optimization algorithms via faster quantum gradient computation. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1425–1444. SIAM, 2019.

22 András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. ACM, June 2019. `doi:10.1145/3313276.3316366`.

23 Michael H. Goerz. *Optimizing robust quantum gates in open quantum systems*. PhD thesis, Universitä"t Kassel, 2015.

24 Vittorio Gorini, Andrzej Kossakowski, and Ennackal Chandy George Sudarshan. Completely positive dynamical semigroups of N-level systems. *Journal of Mathematical Physics*, 17(5):821–825, 1976.

25 Hartmut Häffner, Christian F. Roos, and Rainer Blatt. Quantum computing with trapped ions. *Physics Reports*, 469(4):155–203, 2008.

26 Wenhao He, Tongyang Li, Xiantao Li, Zecheng Li, Chunhao Wang, and Ke Wang. Efficient optimal control of open quantum systems. *arXiv preprint*, 2024. `arXiv:2405.19245`.

27 Chi Jin, Praneeth Netrapalli, and Michael I. Jordan. Accelerated gradient descent escapes saddle points faster than gradient descent, 2017. `arXiv:1711.10456`.

28 Stephen P. Jordan. Fast quantum algorithm for numerical gradient estimation. *Physical Review Letters*, 95(5), July 2005. `doi:10.1103/physrevlett.95.050501`.

29 Navin Khaneja, Timo Reiss, Cindie Kehlet, Thomas Schulte-Herbrüggen, and Steffen J. Glaser. Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms. *Journal of Magnetic Resonance*, 172(2):296–305, 2005.

30 Mária Kieferová, Artur Scherer, and Dominic W. Berry. Simulating the dynamics of time-dependent Hamiltonians with a truncated Dyson series. *Physical Review A*, 99(4):042314, 2019.

31 Christiane P. Koch. Controlling open quantum systems: tools, achievements, and limitations. *Journal of Physics: Condensed Matter*, 28(21):213001, 2016.

32 Christiane P. Koch, Ugo Boscain, Tommaso Calarco, Gunther Dirr, Stefan Filipp, Steffen J. Glaser, Ronnie Kosloff, Simone Montangero, Thomas Schulte-Herbrüggen, Dominique Sugny, and Frank K. Wilhelm. Quantum optimal control in quantum technologies. strategic report on current status, visions and goals for research in europe. *EPJ Quantum Technology*, 9(1):19, 2022.

33 Jr-Shin Li, Justin Ruths, and Dionisis Stefanatos. A pseudospectral method for optimal control of open quantum systems. *The Journal of Chemical Physics*, 131(16), 2009.

34 Jun Li, Xiaodong Yang, Xinhua Peng, and Chang-Pu Sun. Hybrid quantum-classical approach to quantum optimal control. *Physical review letters*, 118(15):150503, 2017.

35 Xiantao Li and Chunhao Wang. Efficient quantum algorithms for quantum optimal control. In *International Conference on Machine Learning*, pages 19982–19994. PMLR, 2023.

36 Xiantao Li and Chunhao Wang. Simulating Markovian open quantum systems using higher-order series expansion. In *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2023.

37 Xiantao Li and Chunhao Wang. Succinct description and efficient simulation of non-Markovian open quantum systems. *Communications in Mathematical Physics*, 401(1):147–183, January 2023. doi:10.1007/s00220-023-04638-4.

38 Goran Lindblad. On the generators of quantum dynamical semigroups. *Communications in Mathematical Physics*, 48(2):119–130, 1976.

39 Jin-Peng Liu and Lin Lin. Dense outputs from quantum simulations. *arXiv preprint*, 2023. arXiv:2307.14441.

40 Seth Lloyd and Simone Montangero. Information theoretical analysis of quantum optimal control. *Physical Review Letters*, 113(1):010502, 2014.

41 Guang Hao Low and Nathan Wiebe. Hamiltonian simulation in the interaction picture. *arXiv preprint*, 2018. arXiv:1805.00675.

42 Alicia B. Magann, Christian Arenz, Matthew D. Grace, Tak-San Ho, Robert L. Kosut, Jarrod R. McClean, Herschel A. Rabitz, and Mohan Sarovar. From pulses to circuits and back again: A quantum optimal control perspective on variational quantum algorithms. *PRX Quantum*, 2(1):010101, 2021.

43 Alicia B Magann, Matthew D Grace, Herschel A Rabitz, and Mohan Sarovar. Digital quantum simulation of molecular dynamics and control. *Physical Review Research*, 3(2):023165, 2021.

44 Yurii Evgen'evich Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Doklady Akademii Nauk*, 269(3):543–547, 1983.

45 José P Palao and Ronnie Kosloff. Optimal control theory for unitary transformations. *Physical Review A*, 68(6):062308, 2003.

46 Daniel M. Reich. *Efficient Characterisation and Optimal Control of Open Quantum Systems-Mathematical Foundations and Physical Applications*. PhD thesis, Universitä"t Kassel, 2015.

47 Yunong Shi, Pranav Gokhale, Prakash Murali, Jonathan M. Baker, Casey Duckering, Yongshan Ding, Natalie C. Brown, Christopher Chamberland, Ali Javadi-Abhari, Andrew W. Cross, David I. Schuster, Kenneth R. Brown, Margaret Martonosi, and Frederic T. Chong. Resource-efficient quantum computing by breaking abstractions. *Proceedings of the IEEE*, 108(8):1353–1370, 2020.

48 Wusheng Zhu, Jair Botina, and Herschel Rabitz. Rapidly convergent iteration methods for quantum optimal control of population. *The Journal of Chemical Physics*, 108(5):1953–1963, 1998.

One-Wayness in Quantum Cryptography

Tomoyuki Morimae
Yukawa Institute for Theoretical Physics, Kyoto University, Japan

Takashi Yamakawa
NTT Social Informatics Laboratories, Tokyo, Japan
NTT Research Center for Theoretical Quantum Information, Atsugi, Japan
Yukawa Institute for Theoretical Physics, Kyoto University, Japan

Abstract

The existence of one-way functions is one of the most fundamental assumptions in classical cryptography. In the quantum world, on the other hand, there are evidences that some cryptographic primitives can exist even if one-way functions do not exist [Kretschmer, TQC 2021; Morimae and Yamakawa, CRYPTO 2022; Ananth, Qian, and Yuen, CRYPTO 2022]. We therefore have the following important open problem in quantum cryptography: What is the most fundamental assumption in quantum cryptography? In this direction, [Brakerski, Canetti, and Qian, ITCS 2023] recently defined a notion called EFI pairs, which are pairs of efficiently generatable states that are statistically distinguishable but computationally indistinguishable, and showed its equivalence with some cryptographic primitives including commitments, oblivious transfer, and general multi-party computations. However, their work focuses on decision-type primitives and does not cover search-type primitives like quantum money and digital signatures. In this paper, we study properties of one-way state generators (OWSGs), which are a quantum analogue of one-way functions proposed by Morimae and Yamakawa. We first revisit the definition of OWSGs and generalize it by allowing mixed output states. Then we show the following results.

1. We define a weaker version of OWSGs, which we call weak OWSGs, and show that they are equivalent to OWSGs. It is a quantum analogue of the amplification theorem for classical weak one-way functions.
2. (Bounded-time-secure) quantum digital signatures with quantum public keys are equivalent to OWSGs.
3. Private-key quantum money schemes (with pure money states) imply OWSGs.
4. Quantum pseudo one-time pad schemes imply both OWSGs and EFI pairs. For EFI pairs, single-copy security suffices.
5. We introduce an incomparable variant of OWSGs, which we call secretly-verifiable and statistically-invertible OWSGs, and show that they are equivalent to EFI pairs.

2012 ACM Subject Classification Theory of computation → Cryptographic primitives

Keywords and phrases Quantum Cryptography

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.4

Related Version *Full Version*: https://arxiv.org/abs/2210.03394

Funding *Tomoyuki Morimae*: JST CREST JPMJCR23I3, JST Moonshot JPMJMS2061-5-1-1, JST FOREST, MEXT QLEAP, the Grant-in Aid for Transformative Research Areas (A) 21H05183, and the Grant-in-Aid for Scientific Research (A) No.22H00522.

1 Introduction

One-way functions (OWFs) are functions that are easy to compute but hard to invert. The existence of OWFs is one of the most fundamental assumptions in classical cryptography. OWFs are equivalent to many cryptographic primitives, such as commitments, digital signatures, pseudorandom generators (PRGs), symmetric-key encryption (SKE), and zero-knowledge, etc. Moreover, almost all other cryptographic primitives, such as collision-resistant

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 4; pp. 4:1–4:21
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

hashes, public-key encryption (PKE), oblivious transfer (OT), multi-party computations (MPCs), etc., imply OWFs. In the quantum world, on the other hand, it seems that OWFs are not necessarily the most fundamental element. In fact, recently, several quantum cryptographic primitives, such as commitments, (one-time secure) digital signatures, quantum pseudo one-time pad (QPOTP)[1], and MPCs are constructed from pseudorandom states generators (PRSGs) [18, 3]. A PRSG [13], which is a quantum analogue of a PRG, is a QPT algorithm that outputs a quantum state whose polynomially-many copies are computationally indistinguishable from the same number of copies of Haar random states. Kretschmer [14] showed that PRSGs exist even if $\mathbf{BQP} = \mathbf{QMA}$ (relative to a quantum oracle), which means that PRSGs (and all the above primitives that can be constructed from PRSGs) could exist even if all quantum-secure (classical) cryptographic primitives including OWFs are broken.[2] Kretschmer, Qian, Sinha, and Tal [15] also showed that 1-PRSGs (which are variants of PRSGs secure against adversaries that get only a single copy of the state) exist even if $\mathbf{NP} = \mathbf{P}$. We therefore have the following important open problem in quantum cryptography:

Question 1: *What is the most fundamental assumption in quantum cryptography?*

In classical cryptography, a pair of PPT algorithms whose output probability distributions are statistically distinguishable but computationally indistinguishable is known to be fundamental. Goldreich [8] showed the equivalence of such a pair to PRGs, which also means the equivalence of such a pair to all cryptographic primitives in Minicrypt [11]. It is natural to consider its quantum analogue: a pair of QPT algorithms whose output quantum states are statistically distinguishable but computationally indistinguishable. In fact, such a pair was implicitly studied in quantum commitments [20]. In the canonical form of quantum commitments [22], computationally hiding and statistically binding quantum commitments are equivalent to such pairs. The importance of such a pair as an independent quantum cryptograpic primitive was pointed out in [20, 4]. In particular, the authors of [4] explicitly defined it as *EFI pairs*,[3] and showed that EFI pairs are implied by several quantum cryptographic primitives such as (semi-honest) quantum OT, (semi-honest) quantum MPCs, and (honest-verifier) quantum computational zero-knowledge proofs. It is therefore natural to ask the following question.

Question 2: *Which other quantum cryptographic primitives imply EFI pairs?*

PRSGs and EFI pairs are "decision type" primitives, which correspond to PRGs in classical cryptography. An example of the other type of primitives, namely, "search type" one in classical cryptography, is OWFs. Recently, a quantum analogue of OWFs, so called one-way states generators (OWSGs), are introduced [18]. A OWSG is a QPT algorithm that, on input a classical bit string (key) k, outputs a quantum state $|\phi_k\rangle$. As the security, we require that it is hard to find k' such that $|\langle\phi_k|\phi_{k'}\rangle|^2$ is non-negligible given polynomially many copies of $|\phi_k\rangle$. The authors showed that OWSGs are implied by PRSGs, and that OWSGs imply (one-time secure) quantum digital signatures with quantum public keys. In classical cryptography, OWFs are connected to many cryptographic primitives. We are therefore interested in the following question.

[1] QPOTP schemes are a one-time-secure SKE with quantum ciphertexts where the key length is shorter than the massage length. (For the definition, see Definition 29.)

[2] If $\mathbf{QMA} = \mathbf{BQP}$, then $\mathbf{NP} \subseteq \mathbf{BQP}$. Because all quantum-secure classical cryptographic primitives are in $\mathbf{NP}$, it means that they are broken by QPT algorithms.

[3] It stands for efficiently samplable, statistically far but computationally indistinguishable pairs of distributions.

Question 3: *Which quantum cryptographic primitives are related to OWSGs?*

In classical cryptography, PRGs (i.e., a decision-type primitive) and OWFs (i.e., a search-type primitive) are equivalent. In quantum cryptography, on the other hand, we do not know whether OWSGs and EFI pairs (or PRSGs) are equivalent or not. We therefore have the following open problem.

Question 4: *Are OWSGs and EFI pairs (or PRSGs) equivalent?*

1.1 Our Results

The study of quantum cryptography with complexity assumptions has became active only very recently, and therefore we do not yet have enough knowledge to answer **Question 1**. However, as an important initial step towards the ultimate goal, we give some answers to other questions above. Our results are summarized as follows. (See also Fig. 1.)

1. We first revisit the definition of OWSGs. In the original definition in [18], output states of OWSGs are assumed to be pure states. Moreover, the verification is done as follows: a bit string k' from the adversary is accepted if and only if the state $|\phi_k\rangle\langle\phi_k|$ is measured in the basis $\{|\phi_{k'}\rangle\langle\phi_{k'}|, I - |\phi_{k'}\rangle\langle\phi_{k'}|\}$, and the first result is obtained. (Note that in classical OWFs, the verification is implicit because it is trivial: just computing $f(x')$ for x' given by the adversary, and check whether it is equal to $f(x)$ or not. However, in the quantum case, we have to explicitly define the verification.) In this paper, to capture more general settings, we generalize the definition of OWSGs by allowing outputs to be mixed states. A non-trivial issue that arises from this modification is that there is no canonical way to verify input-output pairs of OWSGs. To deal with this issue, we include such a verification algorithm as a part of syntax of OWSGs.
2. We show an "amplification theorem" for OWSGs. That is, we define weak OWSGs (wOWSGs), which only requires the adversary's advantage to be $1 - 1/\mathrm{poly}(\lambda)$ instead of $\mathsf{negl}(\lambda)$, and show that a parallel repetition of wOWSGs gives OWSGs. This is an analogue of the equivalence of weak one-way functions and (strong) one-way functions in classical cryptography [23].
3. We show that one-time-secure quantum digital signatures (QDSs) with quantum public keys are equivalent to OWSGs.[4] Moreover, we can generically upgrade one-time-secure QDSs into bounded-time-secure one.[5]
4. We show that private-key quantum money schemes (with pure money states or with verification algorithms that satisfy some symmetry) imply OWSGs.
5. We show that QPOTP schemes imply OWSGs. This in particular means that IND-CPA secure quantum SKE or quantum PKE implies OWSGs.
6. We show that single-copy-secure QPOTP schemes imply EFI pairs. Single-copy-security means that the adversary receives only a single copy of the quantum ciphertext. This in particular means that IND-CPA secure quantum SKE or quantum PKE implies EFI pairs.
7. We introduce an incomparable variant of OWSGs, which we call secretly-verifiable and statistically-invertible OWSGs (SV-SI-OWSGs), and show that SV-SI-OWSGs are equivalent to EFI pairs.

[4] A construction of QDSs from OWSGs was already shown in [18], but in this paper, we generalize the definition of OWSGs, and we give the proof in the new definition.

[5] We thank Or Sattath for asking if we can get (stateless) bounded-time QDSs.

We remark that we consider the generalized definition of OWSGs with mixed state outputs by default. However, all the relationships between OWSGs and other primitives naturally extend to the pure state version if we consider the corresponding pure state variants of the primitives.

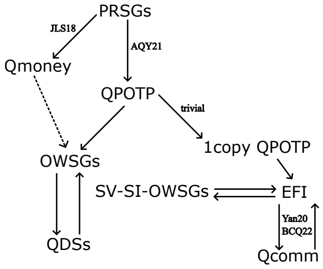


Figure 1 Summary of results. The dotted line means some restrictions: OWSGs are implied by quantum money schemes with *pure* money states or with *symmetric* verification algorithms.

2 Preliminaries

2.1 Basic Notations

We use the standard notations of quantum computing and cryptography. We use λ as the security parameter. $[n]$ means the set $\{1,2,...,n\}$. For any set S, $x \leftarrow S$ means that an element x is sampled uniformly at random from the set S. negl is a negligible function, and poly is a polynomial. PPT stands for (classical) probabilistic polynomial-time and QPT stands for quantum polynomial-time. If we say that an adversary is QPT, it implicitly means non-uniform QPT. A QPT unitary is a unitary operator that can be implemented in a QPT quantum circuit.

For an algorithm A, $y \leftarrow A(x)$ means that the algorithm A outputs y on input x. In particular, if x and y are quantum states and A is a quantum algorithm, $y \leftarrow A(x)$ means the following: a unitary U is applied on $x \otimes |0...0\rangle\langle 0...0|$, and some qubits are traced out. Then, the state of remaining qubits is y. This, importantly, means that the state y is *uniquely decided* by the state x. If A is a QPT algorithm, the unitary U is QPT and the number of ancilla qubits $|0...0\rangle$ is poly(λ). If x is a classical bit string, y is a quantum state, and A is a quantum algorithm, $y \leftarrow A(x)$ sometimes means the following: a unitary U_x that depends on x is applied on $|0...0\rangle$, and some qubits are traced out. The state of the remaining qubits is y. This picture is the same as the most general one where x is given as input, but we sometime choose this picture if it is more convenient.

$\|X\|_1 := \mathrm{Tr}\sqrt{X^\dagger X}$ is the trace norm. $\mathrm{Tr}_{\mathbf{A}}(\rho_{\mathbf{A},\mathbf{B}})$ means that the subsystem (register) $\mathbf{A}$ of the state $\rho_{\mathbf{A},\mathbf{B}}$ on two subsystems (registers) $\mathbf{A}$ and $\mathbf{B}$ is traced out. For simplicity, we sometimes write $\mathrm{Tr}_{\mathbf{A},\mathbf{B}}(|\psi\rangle_{\mathbf{A},\mathbf{B}})$ to mean $\mathrm{Tr}_{\mathbf{A},\mathbf{B}}(|\psi\rangle\langle\psi|_{\mathbf{A},\mathbf{B}})$. I is the two-dimensional identity operator. For simplicity, we sometimes write $I^{\otimes n}$ as I if the dimension is clear from the context. For the notational simplicity, we sometimes write $|0...0\rangle$ just as $|0\rangle$, when the number of zeros is clear from the context. For two pure states $|\psi\rangle$ and $|\phi\rangle$, we sometimes write $\||\psi\rangle\langle\psi| - |\phi\rangle\langle\phi|\|_1$ as $\||\psi\rangle - |\phi\rangle\|_1$ to simplify the notation. $F(\rho,\sigma) := \|\sqrt{\rho}\sqrt{\sigma}\|_1^2$ is the fidelity between ρ and σ. We often use the well-known relation between the trace distance and the fidelity: $1 - \sqrt{F(\rho,\sigma)} \le \frac{1}{2}\|\rho - \sigma\|_1 \le \sqrt{1 - F(\rho,\sigma)}$.

2.2 EFI Pairs

The concept of EFI pairs was implicitly studied in [20], and explicitly defined in [4].

▶ **Definition 1** (EFI pairs [4]). *An EFI pair is an algorithm* $\mathsf{StateGen}(b, 1^\lambda) \to \rho_b$ *that, on input* $b \in \{0,1\}$ *and the security parameter* λ*, outputs a quantum state* ρ_b *such that all of the following three conditions are satisfied.*

- *It is a uniform QPT algorithm.*
- ρ_0 *and* ρ_1 *are computationally indistinguishable. In other words, for any QPT adversary* $\mathcal{A}$, $|\Pr[1 \leftarrow \mathcal{A}(1^\lambda, \rho_0)] - \Pr[1 \leftarrow \mathcal{A}(1^\lambda, \rho_1)]| \le \mathsf{negl}(\lambda)$.
- ρ_0 *and* ρ_1 *are statistically distinguishable, i.e.,* $\frac{1}{2}\|\rho_0 - \rho_1\|_1 \ge \frac{1}{\mathrm{poly}(\lambda)}$.

▶ Remark 2. Note that in the above definition, the statistical distinguishability is defined with only $\ge 1/\mathrm{poly}(\lambda)$ advantage. However, if EFI pairs with the above definition exist, EFI pairs with $\ge 1 - \mathsf{negl}(\lambda)$ statistical distinguishability exist as well. In fact, we have only to define a new $\mathsf{StateGen}'$ that runs $\mathsf{StateGen}$ n times with sufficiently large $n = \mathrm{poly}(\lambda)$, and outputs $\rho_b^{\otimes n}$. The $\ge 1 - \mathsf{negl}(\lambda)$ statistical distinguishability for $\mathsf{StateGen}'$ is shown from the inequality [4], $\frac{1}{2}\|\rho^{\otimes n} - \sigma^{\otimes n}\|_1 \ge 1 - \exp(-n\|\rho - \sigma\|_1/4)$. The computational indistinguishability for $\mathsf{StateGen}'$ is shown by the standard hybrid argument.

2.3 Quantum Commitments

We define canonical quantum bit commitments [20] as follows.

▶ **Definition 3** (Canonical quantum bit commitments [20]). *A canonical quantum bit commitment scheme is a family* $\{Q_0(\lambda), Q_1(\lambda)\}_{\lambda \in \mathbb{N}}$ *of QPT unitaries on two registers* $\mathbf{C}$ *(called the* commitment *register) and* $\mathbf{R}$ *(called the* reveal *register). For simplicity, we often omit* λ *and simply write* $\{Q_0, Q_1\}$ *to mean* $\{Q_0(\lambda), Q_1(\lambda)\}_{\lambda \in \mathbb{N}}$.

▶ Remark 4. Canonical quantum bit commitments are used as follows. In the commit phase, to commit to a bit $b \in \{0,1\}$, the sender generates a state $Q_b|0\rangle_{\mathbf{C},\mathbf{R}}$ and sends $\mathbf{C}$ to the receiver while keeping $\mathbf{R}$. In the reveal phase, the sender sends b and $\mathbf{R}$ to the receiver. The receiver projects the state on $(\mathbf{C}, \mathbf{R})$ onto $Q_b|0\rangle_{\mathbf{C},\mathbf{R}}$, and accepts if it succeeds and otherwise rejects. (In other words, the receiver applies the unitary $Q_b^\dagger$ on the registers $\mathbf{C}$ and $\mathbf{R}$, and measure all qubits in the computational basis. If all result are zero, accept. Otherwise, reject.)

▶ **Definition 5** (Hiding). *We say that a canonical quantum bit commitment scheme* $\{Q_0, Q_1\}$ *is computationally (rep. statistically)* hiding *if* $\mathrm{Tr}_{\mathbf{R}}(Q_0 |0\rangle_{\mathbf{C},\mathbf{R}})$ *is computationally (resp. statistically) indistinguishable from* $\mathrm{Tr}_{\mathbf{R}}(Q_1 |0\rangle_{\mathbf{C},\mathbf{R}})$*. We say that it is perfectly hiding if they are identical states.*

▶ **Definition 6** (Binding). *We say that a canonical quantum bit commitment scheme* $\{Q_0, Q_1\}$ *is computationally (rep. statistically)* binding *if for any QPT (resp. unbounded-time) unitary* U *over* $\mathbf{R}$ *and an additional register* $\mathbf{Z}$ *and any polynomial-size state* $|\tau\rangle_{\mathbf{Z}}$*, it holds that*

$$\left\|(\langle 0| Q_1^\dagger)_{\mathbf{C},\mathbf{R}}(I_{\mathbf{C}} \otimes U_{\mathbf{R},\mathbf{Z}})((Q_0 |0\rangle)_{\mathbf{C},\mathbf{R}} |\tau\rangle_{\mathbf{Z}})\right\| = \mathsf{negl}(\lambda). \tag{1}$$

We say that it is perfectly hiding if the LHS is 0 *for all unbounded-time unitary* U*.* [6]

[6] The above definition is asymmetric for 0 and 1, but it is easy to show that Equation (1) implies

$$\left\|(\langle 0| Q_0^\dagger)_{\mathbf{C},\mathbf{R}}(I_{\mathbf{C}} \otimes U_{\mathbf{R},\mathbf{Z}})((Q_1 |0\rangle)_{\mathbf{C},\mathbf{R}} |\tau\rangle_{\mathbf{Z}})\right\| = \mathsf{negl}(\lambda)$$

for any U and $|\tau\rangle$.

▶ Remark 7. One may think that honest-binding defined above is too weak because it only considers honestly generated commitments. However, somewhat surprisingly, [20] proved that it is equivalent to another binding notion called the *sum-binding* [5].[7] The sum-binding property requires that the sum of probabilities that any (quantum polynomial-time, in the case of computational binding) *malicious* sender can open a commitment to 0 and 1 is at most $1+\mathsf{negl}(\lambda)$. In addition, it has been shown that the honest-binding property is sufficient for cryptographic applications including zero-knowledge proofs/arguments (of knowledge), oblivious transfers, and multi-party computation [22, 6, 18, 21]. In this paper, we refer to honest-binding if we simply write binding.

In this paper, we use the following result.

▶ **Theorem 8** (Converting flavors [20, 10]). *Let $\{Q_0, Q_1\}$ be a canonical quantum bit commitment scheme. Then there exists a canonical quantum bit commitment scheme $\{Q_0', Q_1'\}$, and the following hold for $\mathtt{X}, \mathtt{Y} \in \{computationally, statistically, perfectly\}$:*

- *If $\{Q_0, Q_1\}$ is $\mathtt{X}$ hiding, then $\{Q_0', Q_1'\}$ is $\mathtt{X}$ binding.*
- *If $\{Q_0, Q_1\}$ is $\mathtt{Y}$ binding, then $\{Q_0', Q_1'\}$ is $\mathtt{Y}$ hiding.*

3 OWSGs

In this section, we first define OWSGs (Section 3.1). We then define weak OWSGs and show that weak OWSGs are equivalent to OWSGs (Section 3.2).

3.1 Definition of OWSGs

In this subsection, we define OWSGs. Note that the definition below is a generalization of the one given in [18] in the following three points. First, in [18], the generated states are pure, but here they can be mixed. Second, in [18], the secret key k is uniformly sampled at random, but now it is sampled by a QPT algorithm. Third, in [18], the verification algorithm is the specific algorithm that accepts the alleged key k' with probability $|\langle\phi_k|\phi_{k'}\rangle|^2$, while here we consider a general verification algorithm. We think the definition below is more general (and therefore more fundamental) than that in [18]. Hence hereafter we choose the definition below as the definition of OWSGs.

▶ **Definition 9** (One-way states generators (OWSGs)). *A one-way states generator (OWSG) is a set of algorithms* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *such that*

- $\mathsf{KeyGen}(1^\lambda) \to k$ *: It is a QPT algorithm that, on input the security parameter λ, outputs a classical key $k \in \{0,1\}^\kappa$.*
- $\mathsf{StateGen}(k) \to \phi_k$ *: It is a QPT algorithm that, on input k, outputs an m-qubit quantum state ϕ_k.*
- $\mathsf{Ver}(k', \phi_k) \to \top/\bot$ *: It is a QPT algorithm that, on input ϕ_k and a bit string k', outputs $\top$ or $\bot$.*

We require the following correctness and security.

Correctness: $\Pr\big[\top \leftarrow \mathsf{Ver}(k, \phi_k) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k)\big] \geq 1 - \mathsf{negl}(\lambda)$.

Security: *For any QPT adversary $\mathcal{A}$ and any polynomial t*[8]*,*

$$\Pr\big[\top \leftarrow \mathsf{Ver}(k', \phi_k) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k), k' \leftarrow \mathcal{A}(1^\lambda, \phi_k^{\otimes t})\big] \leq \mathsf{negl}(\lambda).$$

[7] The term "sum-binding" is taken from [19].

[8] $\mathsf{StateGen}$ is actually run t times to generate t copies of ϕ_k, but for simplicity, we just write $\phi_k \leftarrow \mathsf{StateGen}(k)$ only once. This simplification will often be used in this paper.

▶ Remark 10. If ϕ_k is pure, StateGen runs as follows. Apply a QPT unitary U on $|k\rangle|0...0\rangle$ to generate $|\phi_k\rangle \otimes |\eta_k\rangle$, and output $|\phi_k\rangle$. In this case, the existence of the "junk state" $|\eta_k\rangle$ is essential, because otherwise it is not secure against a QPT adversary who does the application of $U^\dagger$ and the computational-basis measurement.

▶ Remark 11. Note that statistically-secure OWSGs do not exist. In other words, there exists an unbounded algorithm $\mathcal{A}$ that can break the security of OWSGs as follows:

1. Given $\phi_k^{\otimes t}$ with a certain polynomial t as input, run the shadow tomography algorithm [1] to find k' such that $\Pr[\mathsf{Ver}(k', \phi_k) \to \top] \geq 1 - \frac{1}{\mathrm{poly}(\lambda)}$. If there exists such k', such k' can be found with only a certain polynomial t. If there is no such k', choose k' uniformly at ramdom.
2. Output k'.

3.2 Hardness Amplification for OWSGs

In this subsection, we define a weaker variant called weak one-way states generators (wOWSGs), and show that they are equivalent to OWSGs.

wOWSGs are defined as follows.

▶ **Definition 12** (Weak one-way states generators (wOWSGs)). *A weak one-way states generator (wOWSG) is a tuple of algorithms* (KeyGen, StateGen, Ver) *defined similarly to OWSGs except that the security is replaced with the following weak security.*

Weak Security: *There exists a polynomial p such that for any QPT adversary $\mathcal{A}$ and any polynomial t,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}(k', \phi_k) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k), k' \leftarrow \mathcal{A}(1^\lambda, \phi_k^{\otimes t})\right] \leq 1 - \frac{1}{p}.$$

We prove that the existence of wOWSGs imply the existence of OWSGs. This is an analogue of Yao's amplification theorem for OWFs in the classical setting [23, 9].

▶ **Theorem 13.** *OWSGs exist if and only if wOWSGs exist.*

For its proof, see the full version.

4 QDSs

In this section, we first define QDSs (Section 4.1), and show that one-time-secure QDSs can be extended to q-time-secure ones (Section 4.2). We then show that one-time-secure QDSs are equivalent to OWSGs (Section 4.3).

4.1 Definition of QDSs

Quantum digital signatures are defined as follows.

▶ **Definition 14** (Quantum digital signatures (QDSs) [18]). *A quantum digital signature (QDS) scheme is a set of algorithms* (SKGen, PKGen, Sign, Ver) *such that*

- $\mathsf{SKGen}(1^\lambda) \to \mathsf{sk}$: *It is a QPT algorithm that, on input the security parameter λ, outputs a classical secret key* sk.
- $\mathsf{PKGen}(\mathsf{sk}) \to \mathsf{pk}$: *It is a QPT algorithm that, on input* sk*, outputs a quantum public key* pk.
- $\mathsf{Sign}(\mathsf{sk}, m) \to \sigma$: *It is a QPT algorithm that, on input* sk *and a message m, outputs a classical signature σ.*
- $\mathsf{Ver}(\mathsf{pk}, m, \sigma) \to \top/\bot$: *It is a QPT algorithm that, on input* pk*, m, and σ, outputs $\top/\bot$.*

We require the correctness and the security as follows.

Correctness: *For any m,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}(\mathsf{pk},m,\sigma) : \begin{array}{r} \mathsf{sk} \leftarrow \mathsf{SKGen}(1^\lambda), \\ \mathsf{pk} \leftarrow \mathsf{PKGen}(\mathsf{sk}), \\ \sigma \leftarrow \mathsf{Sign}(\mathsf{sk},m) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

q-time security: *Let us consider the following security game,* Exp*, between a challenger $\mathcal{C}$ and a QPT adversary $\mathcal{A}$:*

1. *$\mathcal{C}$ runs* $\mathsf{sk} \leftarrow \mathsf{SKGen}(1^\lambda)$.
2. *$\mathcal{C}$ runs* $\mathsf{pk} \leftarrow \mathsf{PKGen}(\mathsf{sk})$ *t times, and sends* $\mathsf{pk}^{\otimes t}$ *to $\mathcal{A}$.*
3. *For $i = 1$ to q, do:*
   - a. *$\mathcal{A}$ sends a message $m^{(i)}$ to $\mathcal{C}$.*
   - b. *$\mathcal{C}$ runs* $\sigma^{(i)} \leftarrow \mathsf{Sign}(\mathsf{sk}, m^{(i)})$*, and sends $\sigma^{(i)}$ to $\mathcal{A}$.*
4. *$\mathcal{A}$ sends σ' and m' to $\mathcal{C}$.*
5. *$\mathcal{C}$ runs* $\mathsf{pk} \leftarrow \mathsf{PKGen}(\mathsf{sk})$ *and* $v \leftarrow \mathsf{Ver}(\mathsf{pk}, m', \sigma')$*. If $m' \notin \{m^{(1)}, \ldots, m^{(q)}\}$ and $v = \top$, the output of the game is 1. Otherwise, the output of the game is 0.*

For any QPT adversary $\mathcal{A}$ and any polynomial t, $\Pr[\mathsf{Exp} = 1] \leq \mathsf{negl}(\lambda)$.

▶ Remark 15. By using the shadow tomography, we can show that statistically-secure QDSs do not exist.

4.2 Extension to q-time Security

▶ **Theorem 16.** *If one-time-secure QDSs exist, then q-time-secure QDSs exist for any polynomial q.*

The idea is similar to the one-time to q-time conversion for attribute-based encryption in [12]. We first consider a scheme where we generate q^2 key pairs of one-time-secure scheme and uniformly chooses one of q^2 signing keys to generate a signature whenever we run the signing algorithm. This scheme is not q-bounded-secure because the probability that the same signing key is used more than once is non-negligible. However, by a simple combinatorial argument, we can upper bound the probability of such a "bad" event by some constant smaller than 1. Thus, by repeating this construction λ times, we can amplify the security to get q-bounded-secure scheme.

For a formal proof, see the full version.

4.3 Equivalence of OWSGs and QDSs

▶ **Theorem 17.** *OWSGs exist if and only if one-time-secure QDSs exist.*

▶ Remark 18. By using the equivalence between OWSGs and wOWSGs (Theorem 13), the result that one-time-secure QDSs imply OWSGs can be improved to a stronger result (with a similar proof) that one-time-secure QDSs with weak security imply OWSGs. Here, the weak security of QDSs means that there exists a polynomial p such that for any QPT adversary $\mathcal{A}$ and any polynomial t, $\Pr[\mathsf{Exp} = 1] \leq 1 - \frac{1}{p}$.

It is proven in [18] that OWSGs implies one-time-secure QDSs. However, since we generalize the definition of OWSGs, we need to reprove it. Fortunately, almost the same construction as that in [18] works with the generalized definition of OWSGs. Roughly, the construction is as follows when the message space is one-bit: a secret key is $\mathsf{sk} = (k_0, k_1)$, a public key is $\mathsf{pk} = (\phi_{k_0}, \phi_{k_1})$, and a signature for a bit $b \in \{0,1\}$ is k_b. The verification algorithm of QDSs simply runs that of the OWSG.

For the other direction, we construct OWSGs from QDSs by regarding sk and pk of QDSs as k and ϕ_k of OWSGs.

For a formal proof, see the full version.

5 Quantum Money

In this section, we first define private-key quantum money schemes (Section 5.1). We then construct OWSGs from quantum money schemes with pure money states (Section 5.2). We also show that OWSGs can be constructed from quantum money schemes where the verification algorithms satisfy a certain symmetric property (Section 5.3).

5.1 Definition of Private-key Quantum Money

Private-key quantum money schemes are defined as follows.

▶ **Definition 19** (Private-key quantum money [13, 2]). *A private-key quantum money scheme is a set of algorithms* $(\mathsf{KeyGen},\mathsf{Mint},\mathsf{Ver})$ *such that*

- $\mathsf{KeyGen}(1^\lambda)\to k$ *: It is a QPT algorithm that, on input the security parameter* λ*, outputs a classical secret key* k*.*
- $\mathsf{Mint}(k)\to \$_k$ *: It is a QPT algorithm that, on input* k*, outputs an* m*-qubit quantum state* $\$_k$*.*
- $\mathsf{Ver}(k,\rho)\to\top/\bot$ *: It is a QPT algorithm that, on input* k *and a quantum state* ρ*, outputs* $\top/\bot$*.*

We require the following correctness and security.

Correctness:

$$\Pr\left[\top\leftarrow\mathsf{Ver}(k,\$_k):k\leftarrow\mathsf{KeyGen}(1^\lambda),\$_k\leftarrow\mathsf{Mint}(k)\right]\ge 1-\mathsf{negl}(\lambda).$$

Security: *For any QPT adversary* $\mathcal{A}$ *and any polynomial* t*,*

$$\Pr\left[\mathsf{Count}(k,\xi)\ge t+1:k\leftarrow\mathsf{KeyGen}(1^\lambda),\$_k\leftarrow\mathsf{Mint}(k),\xi\leftarrow\mathcal{A}(1^\lambda,\$_k^{\otimes t})\right]\le\mathsf{negl}(\lambda),$$

where ξ *is a quantum state on* ℓ *registers,* $R_1,...,R_\ell$*, each of which is of* m *qubits, and* Count *is the following QPT algorithm: on input* ξ*, it runs* $\top/\bot\leftarrow\mathsf{Ver}(k,\xi_j)$ *for each* $j\in[1,2,...,\ell]$*, where* $\xi_j:=Tr_{R_1,...,R_{j-1},R_{j+1},...,R_\ell}(\xi)$*, and outputs the total number of* $\top$*.*

▶ Remark 20. Private-key quantum money schemes are constructed from PRSGs [13].

▶ Remark 21. As is shown in [1], private-key quantum money schemes are broken by an unbounded adversary, and therefore statistically-secure private-key quantum money schemes do not exist. (The idea is as follows: the unbounded adversary first finds all $\{k_i\}_i$ such that $\mathsf{Ver}(k_i,\$_k)$ is large with the shadow tomography, and then searches a state ρ by the brute-force such that $\mathsf{Ver}(k_i,\rho)$ is close to $\mathsf{Ver}(k_i,\$_k)$ FOR ALL i. Finally, the adversary outputs many copies of ρ.)

5.2 OWSGs from Quantum Money with Pure Money States

▶ **Theorem 22.** *If private-key quantum money schemes with pure quantum money states exist, then OWSGs exist.*

▶ Remark 23. For example, the private-key quantum money scheme of [13] has pure quantum money states.

▶ Remark 24. By using the equivalence between OWSGs and wOWSGs (Theorem 13), this result can be improved to a stronger result (with a similar proof) that private-key quantum money schemes with pure quantum money states and with weak security imply OWSGs. Here, the weak security means that there exists a polynomial p such that for any QPT adversary $\mathcal{A}$ and any polynomial t,

$$\Pr\left[\mathsf{Count}(k,\xi)\ge t+1:k\leftarrow\mathsf{KeyGen}(1^\lambda),\$_k\leftarrow\mathsf{Mint}(k),\xi\leftarrow\mathcal{A}(1^\lambda,\$_k^{\otimes t})\right]\le 1-\frac{1}{p}.$$

Proof of Theorem 22. Let $(\mathsf{QM.KeyGen}, \mathsf{QM.Mint}, \mathsf{QM.Ver})$ be a private-key quantum money scheme with pure money states. From it, we construct a OWSG as follows.

- $\mathsf{KeyGen}(1^\lambda) \to k$: Run $k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)$. Output k.
- $\mathsf{StateGen}(k) \to \phi_k$: Run $|\$_k\rangle \leftarrow \mathsf{QM.Mint}(k)$. Output $\phi_k := |\$_k\rangle\langle\$_k|$.
- $\mathsf{Ver}(k', \phi_k) \to \top/\bot$: Parse $\phi_k = |\$_k\rangle\langle\$_k|$. Measure $|\$_k\rangle$ with the basis $\{|\$_{k'}\rangle\langle\$_{k'}|, I - |\$_{k'}\rangle\langle\$_{k'}|\}$, and output $\top$ if the first result is obtained. Output $\bot$ if the second result is obtained. (This measurement is done in the following way: generate $U(|k'\rangle|0...0\rangle) = |\$_{k'}\rangle|\eta_{k'}\rangle$, and discard the first register. Then apply $U^\dagger$ on $|\$_k\rangle|\eta_{k'}\rangle$, and measure all qubits in the computationl basis. If the result is $k'0...0$, accept. Otherwise, reject.)

The correctness is clear. Let us show the security. Assume that it is not secure. Then, there exists a QPT adversary $\mathcal{A}$, a polynomial t, and a polynomial p such that

$$\sum_{k,k'} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr\big[k' \leftarrow \mathcal{A}(|\$_k\rangle^{\otimes t})\big] |\langle\$_k|\$_{k'}\rangle|^2 \geq \frac{1}{p}.$$

Define the set $S := \Big\{(k,k') \;\Big|\; |\langle\$_k|\$_{k'}\rangle|^2 \geq \frac{1}{2p}\Big\}$. Then, we have

$$\sum_{(k,k')\in S} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr\big[k' \leftarrow \mathcal{A}(|\$_k\rangle^{\otimes t})\big] > \frac{1}{2p}.$$

This is shown as follows.

$$\begin{aligned}
\frac{1}{p} &\leq \sum_{k,k'} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr\big[k' \leftarrow \mathcal{A}(|\$_k\rangle^{\otimes t})\big] |\langle\$_k|\$_{k'}\rangle|^2 \\
&= \sum_{(k,k')\in S} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr\big[k' \leftarrow \mathcal{A}(|\$_k\rangle^{\otimes t})\big] |\langle\$_k|\$_{k'}\rangle|^2 \\
&\quad + \sum_{(k,k')\notin S} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr\big[k' \leftarrow \mathcal{A}(|\$_k\rangle^{\otimes t})\big] |\langle\$_k|\$_{k'}\rangle|^2 \\
&< \sum_{(k,k')\in S} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr\big[k' \leftarrow \mathcal{A}(|\$_k\rangle^{\otimes t})\big] + \frac{1}{2p}.
\end{aligned}$$

Let us also define $T := \Big\{k \;\Big|\; \Pr[\top \leftarrow \mathsf{QM.Ver}(k, |\$_k\rangle)] \geq 1 - \frac{1}{8p}\Big\}$. Then, $\sum_{k\in T} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] > 1 - \mathsf{negl}(\lambda)$. This is shown as follows.

$$\begin{aligned}
1 - \mathsf{negl}(\lambda) &\leq \sum_{k} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr[\top \leftarrow \mathsf{QM.Ver}(k, |\$_k\rangle)] \\
&= \sum_{k\in T} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr[\top \leftarrow \mathsf{QM.Ver}(k, |\$_k\rangle)] \\
&\quad + \sum_{k\notin T} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \Pr[\top \leftarrow \mathsf{QM.Ver}(k, |\$_k\rangle)] \\
&< \sum_{k\in T} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big] \\
&\quad + \Big(1 - \frac{1}{8p}\Big)\Big(1 - \sum_{k\in T} \Pr\big[k \leftarrow \mathsf{QM.KeyGen}(1^\lambda)\big]\Big).
\end{aligned}$$

Here, the first inequality is from the correctness of the quantum money scheme.

Let us fix (k,k') such that $(k,k')\in S$ and $k\in T$. The probability of having such (k,k') is, from the union bound,

$$\begin{aligned}\sum_{(k,k')\in S\wedge k\in T}\Pr\big[k\leftarrow\mathsf{QM.KeyGen}(1^\lambda)\big]\Pr\big[k'\leftarrow\mathcal{A}(|\$_k\rangle^{\otimes t})\big] &> \frac{1}{2p}+1-\mathsf{negl}(\lambda)-1\\ &= \frac{1}{2p}-\mathsf{negl}(\lambda).\end{aligned}$$

From the $\mathcal{A}$, we construct a QPT adversary $\mathcal{B}$ that breaks the security of the private-key quantum money scheme as follows: On input $|\$_k\rangle^{\otimes t}$, it runs $k'\leftarrow\mathcal{A}(|\$_k\rangle^{\otimes t})$. It then runs $|\$_{k'}\rangle\leftarrow\mathsf{QM.Mint}(k')$ ℓ times, where ℓ is a polynomial specified later, and outputs $\xi:=|\$_{k'}\rangle^{\otimes\ell}$. Let us show that thus defined $\mathcal{B}$ breaks the security of the private-key quantum money scheme. Let v_j be the bit that is 1 if the output of $\mathsf{QM.Ver}(k,\xi_j)$ is $\top$, and is 0 otherwise. Then, for any (k,k') such that $(k,k')\in S$ and $k\in T$,

$$\begin{aligned}\Pr[v_j=1] &= \Pr[\top\leftarrow\mathsf{QM.Ver}(k,\xi_j)]=\Pr[\top\leftarrow\mathsf{QM.Ver}(k,|\$_{k'}\rangle)]\\ &\ge \Pr[\top\leftarrow\mathsf{QM.Ver}(k,|\$_k\rangle)]-\sqrt{1-\frac{1}{2p}}\ge 1-\frac{1}{8p}-\sqrt{1-\frac{1}{2p}}\ge\frac{1}{8p}\end{aligned}$$

for each $j\in[1,2,...,\ell]$. Here, in the first inequality, we have used the fact that $\Pr[1\leftarrow\mathcal{D}(|\$_k\rangle)]-\Pr[1\leftarrow\mathcal{D}(|\$_{k'}\rangle)]\le\sqrt{1-\frac{1}{2p}}$ for any algorithm $\mathcal{D}$. This is because $|\langle\$_k|\$_{k'}\rangle|^2\ge\frac{1}{2p}$ for any $(k,k')\in S$.[9] Moreover, in the second inequality, we have used the fact that $\Pr[\top\leftarrow\mathsf{QM.Ver}(k,|\$_k\rangle)]\ge1-\frac{1}{8p}$ for any $k\in T$. Finally, in the last inequality, we have used the Bernoulli's inequality.[10]

Let us take $\ell\ge\max(16p(t+1),16^2p^3)$. Then, for any (k,k') such that $(k,k')\in S$ and $k\in T$,

$$\begin{aligned}\Pr\big[\mathsf{Count}(k,|\$_{k'}\rangle^{\otimes\ell})\ge t+1\big] &= \Pr\left[\sum_{j=1}^{\ell}v_j\ge t+1\right]\ge\Pr\left[\sum_{j=1}^{\ell}v_j\ge\frac{\ell}{16p}\right]\\ &= \Pr\left[\sum_{j=1}^{\ell}v_j\ge\frac{\ell}{8p}-\frac{\ell}{16p}\right]\ge\Pr\left[\sum_{j=1}^{\ell}v_j\ge\mathbb{E}(\sum_{j=1}^{\ell}v_j)-\frac{\ell}{16p}\right]\\ &\ge 1-2\exp\left[-\frac{2\ell}{16^2p^2}\right]\ge1-2e^{-2p}.\end{aligned}$$

Here, in the third inequality, we have used Hoeffding's inequality. The probability that $\mathcal{B}$ breaks the security of the quantum money scheme is therefore

$$\begin{aligned}&\sum_{k,k'}\Pr\big[k\leftarrow\mathsf{QM.KeyGen}(1^\lambda)\big]\Pr\big[k'\leftarrow\mathcal{A}(|\$_k\rangle^{\otimes t})\big]\Pr\big[\mathsf{Count}(k,|\$_{k'}\rangle^{\otimes\ell})\ge t+1\big]\\ \ge&\sum_{(k,k')\in S\wedge k\in T}\Pr\big[k\leftarrow\mathsf{QM.KeyGen}(1^\lambda)\big]\Pr\big[k'\leftarrow\mathcal{A}(|\$_k\rangle^{\otimes t})\big]\Pr\big[\mathsf{Count}(k,|\$_{k'}\rangle^{\otimes\ell})\ge t+1\big]\\ \ge&(1-2e^{-2p})\sum_{(k,k')\in S\wedge k\in T}\Pr\big[k\leftarrow\mathsf{QM.KeyGen}(1^\lambda)\big]\Pr\big[k'\leftarrow\mathcal{A}(|\$_k\rangle^{\otimes t})\big]\\ \ge&(1-2e^{-2p})\Big(\frac{1}{2p}-\mathsf{negl}(\lambda)\Big),\end{aligned}$$

which is non-negligible. The $\mathcal{B}$ therefore breaks the security of the private-key quantum money scheme. ◀

[9] Due to the relation between the fidelity and the trace distance, we have $\frac{1}{2}\||\$_k\rangle\langle\$_k|-|\$_{k'}\rangle\langle\$_{k'}|\|_1\le\sqrt{1-|\langle\$_k|\$_{k'}\rangle|^2}$, which means that $\langle\$_k|\Pi|\$_k\rangle-\langle\$_{k'}|\Pi|\$_{k'}\rangle\le\sqrt{1-|\langle\$_k|\$_{k'}\rangle|^2}$ for any POVM element Π.

[10] $(1+x)^r\le1+rx$ for any real r and x such that $0\le r\le1$ and $x\ge-1$.

5.3 OWSGs from Quantum Money with Symmetric Verifiability

We consider the following restriction for quantum money.

▶ **Definition 25** (Symmetric-verifiability)**.** *We say that a private-key quantum money scheme satisfies the symmetric-verifiability if* $\Pr[\top \leftarrow \mathsf{Ver}(k,\$_{k'})] = \Pr[\top \leftarrow \mathsf{Ver}(k',\$_k)]$ *for all* $k \neq k'$*.*

▶ Remark 26. For example, if all money states are pure, and $\mathsf{Ver}(\alpha,\rho)$ is the following algorithm, the symmetric-verifiability is satisfied: Measure ρ with the basis $\{|\$_\alpha\rangle\langle\$_\alpha|, I-|\$_\alpha\rangle\langle\$_\alpha|\}$. If the first result is obtained, output $\top$. Otherwise, output $\bot$.

▶ **Theorem 27.** *If private-key quantum money schemes with symmetric-verifiability exist, then OWSGs exist.*

▶ Remark 28. By using the equivalence between OWSGs and wOWSGs (Theorem 13), this result can be improved to a stronger result (with a similar proof) that private-key quantum money schemes with symmetric-verifiability and with weak security imply OWSGs. Here, the weak security means that there exists a polynomial p such that for any QPT adversary $\mathcal{A}$ and any polynomial t,

$$\Pr\left[\mathsf{Count}(k,\xi) \geq t+1 : k \leftarrow \mathsf{KeyGen}(1^\lambda), \$_k \leftarrow \mathsf{Mint}(k), \xi \leftarrow \mathcal{A}(1^\lambda, \$_k^{\otimes t})\right] \leq 1 - \frac{1}{p}.$$

The proof of Theorem 27 is similar to that of Theorem 22. For a proof, see the full version.

6 QPOTP

In this section, we first define (IND-based) QPOTP schemes (Section 6.1). We then show that QPOTP schemes imply OWSGs (Section 6.2), and that single-copy-secure QPOTP schemes imply EFI pairs (Section 6.3).

6.1 Definition of QPOTP

Quantum pseudo one-time pad schemes are defined as follows.

▶ **Definition 29** ((IND-based) quantum pseudo one-time pad (QPOTP))**.** *An (IND-based) quantum pseudo one-time pad (QPOTP) scheme with the key length* κ *and the plaintext length* ℓ $(\ell > \kappa)$ *is a set of algorithms* $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *such that*

- $\mathsf{KeyGen}(1^\lambda) \to \mathsf{sk}$: *It is a QPT algorithm that, on input the security parameter* λ*, outputs a classical secret key* $\mathsf{sk} \in \{0,1\}^\kappa$*.*
- $\mathsf{Enc}(\mathsf{sk},x) \to \mathsf{ct}$: *It is a QPT algorithm that, on input* sk *and a classical plaintext message* $x \in \{0,1\}^\ell$*, outputs an* ℓn*-qubit quantum ciphertext* ct*.*
- $\mathsf{Dec}(\mathsf{sk},\mathsf{ct}) \to x'$: *It is a QPT algorithm that, on input* sk *and* ct*, outputs* $x' \in \{0,1\}^\ell$*.*

We require the following correctness and security.

Correctness: *For any* $x \in \{0,1\}^\ell$, $\Pr\left[x \leftarrow \mathsf{Dec}(\mathsf{sk},\mathsf{ct}) : \mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda), \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{sk},x)\right] \geq 1-\mathsf{negl}(\lambda)$.

Security: *For any* $x_0, x_1 \in \{0,1\}^\ell$*, any QPT adversary* $\mathcal{A}$*, and any polynomial* t*,*

$$\begin{aligned}&|\Pr\left[1 \leftarrow \mathcal{A}(\mathsf{ct}_0^{\otimes t}) : \mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda), \mathsf{ct}_0 \leftarrow \mathsf{Enc}(\mathsf{sk},x_0)\right]\\ &-\Pr\left[1 \leftarrow \mathcal{A}(\mathsf{ct}_1^{\otimes t}) : \mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda), \mathsf{ct}_1 \leftarrow \mathsf{Enc}(\mathsf{sk},x_1)\right]| \leq \mathsf{negl}(\lambda).\end{aligned}$$

▶ **Definition 30.** *We say that a QPOTP scheme is single-copy-secure if the security holds only for $t = 1$.*

▶ Remark 31. Note that the above definition of QPOTP is different from that of [3] in the following two points. First, we consider a general secret key generation QPT algorithm, while they consider uniform sampling of the secret key. Second, we consider the IND-based version of the security, while the security definition of [3] is as follows: For any $x \in \{0,1\}^\ell$, any QPT adversary $\mathcal{A}$, and any polynomial t,

$$\begin{aligned}&|\Pr[1 \leftarrow \mathcal{A}(\mathsf{ct}^{\otimes t}) : \mathsf{sk} \leftarrow \{0,1\}^\kappa, \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{sk}, x)]\\ &- \Pr[1 \leftarrow \mathcal{A}((|\psi_1\rangle \otimes ... \otimes |\psi_\ell\rangle)^{\otimes t}) : |\psi_1\rangle, ..., |\psi_\ell\rangle \leftarrow \mu_n]| \le \mathsf{negl}(\lambda),\end{aligned}$$

where $|\psi\rangle \leftarrow \mu_n$ means the Haar random sampling of n-qubit states. It is clear that the security definition of [3] implies our IND-based security, and therefore if QPOTP schemes of [3] exist, those of Definition 29 exist. Since our results are constructions of OWSGs and EFI pairs from QPOTP, the above modification only makes our results stronger.

▶ Remark 32. QPOTP is constructed from PRSGs [3].

6.2 OWSGs from QPOTP

▶ **Theorem 33.** *If QPOTP schemes with $\kappa < \ell$ exist, then OWSGs exist.*

Proof of Theorem 33. Let $(\mathsf{OTP.KeyGen}, \mathsf{OTP.Enc}, \mathsf{OTP.Dec})$ be a QPOTP scheme with $\kappa < \ell$. From it, we construct a wOWSG as follows.[11] (From Theorem 13, it is enough for the existence of OWSGs.)

- $\mathsf{KeyGen}(1^\lambda) \to k$: Run $\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda)$. Choose $x \leftarrow \{0,1\}^\ell$. Output $k := (\mathsf{sk}, x)$.
- $\mathsf{StateGen}(k) \to \phi_k$: Parse $k = (\mathsf{sk}, x)$. Run $\mathsf{ct}_{\mathsf{sk},x} \leftarrow \mathsf{OTP.Enc}(\mathsf{sk}, x)$. Output $\phi_k := \mathsf{ct}_{\mathsf{sk},x} \otimes |x\rangle\langle x|$.
- $\mathsf{Ver}(k', \phi_k) \to \top/\bot$: Parse $k' = (\mathsf{sk}', x')$. Parse $\phi_k = \mathsf{ct}_{\mathsf{sk},x} \otimes |x\rangle\langle x|$. Run $x'' \leftarrow \mathsf{OTP.Dec}(\mathsf{sk}', \mathsf{ct}_{\mathsf{sk},x})$. If $x'' = x' = x$, output $\top$. Otherwise, output $\bot$.

The correctness is clear. Let us show the security. Assume that it is not secure. It means that for any polynomial p there exist a QPT adversary $\mathcal{A}$ and a polynomial t such that

$$\Pr\left[x' = x'' = x : \begin{array}{l}\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda),\\ x \leftarrow \{0,1\}^\ell,\\ \mathsf{ct}_{\mathsf{sk},x} \leftarrow \mathsf{OTP.Enc}(\mathsf{sk}, x),\\ (\mathsf{sk}', x') \leftarrow \mathcal{A}(\mathsf{ct}_{\mathsf{sk},x}^{\otimes t} \otimes |x\rangle\langle x|^{\otimes t})\\ x'' \leftarrow \mathsf{OTP.Dec}(\mathsf{sk}', \mathsf{ct}_{\mathsf{sk},x})\end{array}\right] \ge 1 - \frac{1}{p}. \tag{2}$$

From this $\mathcal{A}$, we construct a QPT adversary $\mathcal{B}$ that breaks the security of the QPOTP scheme as follows. Let $b \in \{0,1\}$ be the parameter of the following security game.

1. $\mathcal{B}$ chooses $x_0, x_1 \leftarrow \{0,1\}^\ell$, and sends them to the challenger $\mathcal{C}$.
2. $\mathcal{C}$ runs $\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda)$.
3. $\mathcal{C}$ runs $\mathsf{ct}_{\mathsf{sk},x_b} \leftarrow \mathsf{OTP.Enc}(\mathsf{sk}, x_b)$ $t + 1$ times.
4. $\mathcal{C}$ sends $\mathsf{ct}_{\mathsf{sk},x_b}^{\otimes t+1}$ to $\mathcal{B}$.
5. $\mathcal{B}$ runs $(\mathsf{sk}', x') \leftarrow \mathcal{A}(\mathsf{ct}_{\mathsf{sk},x_b}^{\otimes t} \otimes |x_0\rangle\langle x_0|^{\otimes t})$.
6. $\mathcal{B}$ runs $x'' \leftarrow \mathsf{OTP.Dec}(\mathsf{sk}', \mathsf{ct}_{k,x_b})$. If $x' = x'' = x_0$, $\mathcal{B}$ outputs $b' = 0$. Otherwise, it outputs $b' = 1$.

[11] A similar proof idea was given in Lemma 4.6 of [7]. However, the direct application of the proof will not work, because ciphertexts (and therefore output states of OWSGs) are quantum and the verification of "preimages" is done by the additional verification algorithm.

It is clear that $\Pr[b' = 0|b = 0]$ is equivalent to the left-hand-side of Eq. (2). On the other hand,

$$\begin{aligned}
\Pr\Big[b' = 0|b = 1\Big] &= \frac{1}{2^{2\ell}} \sum_{x_0,x_1,\mathsf{sk},\mathsf{sk}'} \Pr\Big[\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda)\Big] \Pr\Big[\mathsf{sk}' \leftarrow \mathcal{A}(\mathsf{ct}_{\mathsf{sk},x_1}^{\otimes t} \otimes |x_0\rangle\langle x_0|^{\otimes t})\Big] \\
&\quad \times \Pr\Big[x_0 \leftarrow \mathsf{OTP.Dec}(\mathsf{sk}', \mathsf{ct}_{\mathsf{sk},x_1})\Big] \\
&\leq \frac{1}{2^{2\ell}} \sum_{x_0,x_1,\mathsf{sk},\mathsf{sk}'} \Pr\Big[\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda)\Big] \Pr\Big[x_0 \leftarrow \mathsf{OTP.Dec}(\mathsf{sk}', \mathsf{ct}_{\mathsf{sk},x_1})\Big] \\
&= \frac{1}{2^{2\ell}} \sum_{x_1,\mathsf{sk},\mathsf{sk}'} \Pr\Big[\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda)\Big] \sum_{x_0} \Pr\Big[x_0 \leftarrow \mathsf{OTP.Dec}(\mathsf{sk}', \mathsf{ct}_{\mathsf{sk},x_1})\Big] \\
&= \frac{1}{2^{2\ell}} \sum_{x_1,\mathsf{sk},\mathsf{sk}'} \Pr\Big[\mathsf{sk} \leftarrow \mathsf{OTP.KeyGen}(1^\lambda)\Big] = \frac{2^\kappa}{2^\ell} \leq \frac{1}{2}.
\end{aligned}$$

Therefore $|\Pr[b' = 0|b = 0] - \Pr[b' = 0|b = 1]|$ is non-negligible, which means that the $\mathcal{B}$ breaks the security of the QPOTP. ◀

6.3 EFI Pairs from Single-Copy-Secure QPOTP

▶ **Theorem 34.** *If single-copy-secure QPOTP schemes with $\kappa < \ell$ exist then EFI pairs exist.*

We prove this theorem based on a result shown by Lai and Chung [16], which gives a quantum analogue of Shannon's impossibility. Roughly speaking, they show that if a SKE scheme for n-qubit messages and κ-bit secret keys is information theoretically one-time-secure, then we must have $\kappa \geq 2n$. By a reduction to their result via a hybrid encryption of QPOTP and quantum one-time pads, we can show that any QPOTP scheme with $\kappa < \ell$ is *not* one-time-secure against unbounded-time adversaries. On the other hand, we assume that it is one-time-secure against QPT adversaries. This computationally-secure and information-theoretically-insecure encryption scheme can be directly used to construct EFI pairs. For a formal proof, see the full version.

7 SV-SI-OWSGs

In this section, we define SV-SI-OWSGs (Section 7.1), and show that SV-SI-OWSGs are equivalent to EFI pairs (Section 7.2). In Section 7.1, before defining SV-SI-OWSGs, we first define SV-OWSGs for a didactic purpose. We will point out that SV-OWSGs seem to need a more constraint so that they become equivalent to EFI. We then define SV-SI-OWSGs.

7.1 Definition of SV-SI-OWSGs

We first define secretly-verifiable OWSGs (SV-OWSGs) as follows.

▶ **Definition 35** (Secretly-verifiable OWSGs (SV-OWSGs)). *A secretly-verifiable OWSG (SV-OWSG) is a set of algorithms* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *as follows.*

- $\mathsf{KeyGen}(1^\lambda) \to k$: *It is a QPT algorithm that, on input the security parameter λ, outputs a key $k \in \{0,1\}^\kappa$.*
- $\mathsf{StateGen}(k) \to \phi_k$: *It is a QPT algorithm that, on input k, outputs an m-qubit state ϕ_k.*
- $\mathsf{Ver}(k', k) \to \top/\bot$: *It is a QPT algorithm that, on input k and k', outputs $\top/\bot$.*

We require the following two properties.

Correctness:

$$\Pr\left[\top \leftarrow \mathsf{Ver}(k,k) : k \leftarrow \mathsf{KeyGen}(1^\lambda)\right] \geq 1 - \mathsf{negl}(\lambda).$$

Security: *For any QPT adversary $\mathcal{A}$ and any polynomial t,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}(k',k) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k), k' \leftarrow \mathcal{A}(\phi_k^{\otimes t})\right] \leq \mathsf{negl}(\lambda).$$

The following lemma shows that, without loss of generality, Ver can be replaced with the algorithm of just checking whether $k = k'$ or not.

▶ **Lemma 36.** *Let* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *be a SV-OWSG. Then, the following SV-OWSG* $(\mathsf{KeyGen}', \mathsf{StateGen}', \mathsf{Ver}')$ *exists.*

- KeyGen' *and* $\mathsf{StateGen}'$ *are the same as* KeyGen *and* $\mathsf{StateGen}$*, respectively.*
- $\mathsf{Ver}'(k',k) \to \top/\bot$ *: On input k and k', output $\top$ if $k = k'$. Otherwise, output $\bot$.*

For a proof, see the full version.

Note that statistically-secure SV-OWSGs are easy to realize. For example, consider the following construction:

- $\mathsf{KeyGen}(1^\lambda)$: Sample $k \leftarrow \{0,1\}^\lambda$.
- $\mathsf{StateGen}(k)$: Output $\frac{I^{\otimes m}}{2^m}$.
- $\mathsf{Ver}(k',k)$: Output $\top$ if $k' = k$. Otherwise, output $\bot$.

We therefore need a constraint to have a meaningful primitive. We define secretly-verifiable and statistically-invertible OWSGs (SV-SI-OWSGs) as follows. Introducing the statistical invertibility allows us to avoid trivial constructions with the statistical security.

▶ **Definition 37** (Secretly-verifiable and statistically-invertible OWSGs (SV-SI-OWSGs))**.** *A secretly-verifiable and statistically-invertible OWSG (SV-SI-OWSG) is a set of algorithms* $(\mathsf{KeyGen}, \mathsf{StateGen})$ *as follows.*

- $\mathsf{KeyGen}(1^\lambda) \to k$ *: It is a QPT algorithm that, on input the security parameter λ, outputs a key $k \in \{0,1\}^\kappa$.*
- $\mathsf{StateGen}(k) \to \phi_k$ *: It is a QPT algorithm that, on input k, outputs an m-qubit state ϕ_k.*

We require the following two properties.

Statistical invertibility: *There exists a polynomial p such that, for any k and k' $(k \neq k')$,* $\frac{1}{2}\|\phi_k - \phi_{k'}\|_1 \geq \frac{1}{p}$.

Computational non-invertibility: *For any QPT adversary $\mathcal{A}$ and any polynomial t,*

$$\Pr\left[k \leftarrow \mathcal{A}(\phi_k^{\otimes t}) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k)\right] \leq \mathsf{negl}(\lambda).$$

The following lemma shows that the statistical invertibility with advantage $\frac{1}{\mathrm{poly}(\lambda)}$ can be amplified to $1 - 2^{-q}$ for any polynomial q.

▶ **Lemma 38.** *If a SV-SI-OWSG exists then a SV-SI-OWSG with statistical invertibility larger than $1 - 2^{-q}$ with any polynomial q exists.*

Proof. Let $(\mathsf{KeyGen}, \mathsf{StateGen})$ be a SV-SI-OWSG with statistical invertibility larger than $\frac{1}{p}$, where p is a polynomial. From it, we construct a new SV-SI-OWSG $(\mathsf{KeyGen}', \mathsf{StateGen}')$ as follows:

- $\mathsf{KeyGen}'(1^\lambda) \to k$: Run $k \leftarrow \mathsf{KeyGen}(1^\lambda)$, and output k.
- $\mathsf{StateGen}'(k) \to \phi'_k$: Run $\phi_k \leftarrow \mathsf{StateGen}(k)$ $2pq$ times, and output $\phi'_k := \phi_k^{\otimes 2pq}$.

First, for any k and k' ($k \neq k'$),

$$\begin{aligned}\frac{1}{2}\|\phi'_k - \phi'_{k'}\|_1 &= \frac{1}{2}\|\phi_k^{\otimes 2pq} - \phi_{k'}^{\otimes 2pq}\|_1 \geq 1 - \exp(-2qp\|\phi_k - \phi_{k'}\|_1/4)\\ &\geq 1 - \exp(-q) \geq 1 - 2^{-q},\end{aligned}$$

which shows the statistical invertibility of $(\mathsf{KeyGen}', \mathsf{StateGen}')$ with the advantage larger than $1 - 2^{-q}$. Second, from the computational non-invertibility of $(\mathsf{KeyGen}, \mathsf{StateGen})$,

$$\begin{aligned}&\Pr\big[k \leftarrow \mathcal{A}(\phi_k'^{\otimes t}) : k \leftarrow \mathsf{KeyGen}'(1^\lambda), \phi'_k \leftarrow \mathsf{StateGen}'(k)\big]\\ =\ &\Pr\Big[k \leftarrow \mathcal{A}(\phi_k^{\otimes 2pqt}) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k)\Big] \leq \mathsf{negl}(\lambda)\end{aligned}$$

for any QPT adversary $\mathcal{A}$ and any polynomial t, which shows the computational non-invertibility of $(\mathsf{KeyGen}', \mathsf{StateGen}')$. ◀

The following lemma shows that the statistical invertibility is equivalent to the existence of a (unbounded) adversary that can find the correct k given many copies of ϕ_k except for a negligible error.

▶ **Lemma 39.** *The statistical invertibility is satisfied if and only if the following is satisfied: There exists a (not necessarily QPT) POVM measurement $\{\Pi_k\}_{k\in\{0,1\}^\kappa}$ and a polynomial t such that $Tr(\Pi_k\phi_k^{\otimes t}) \geq 1 - \mathsf{negl}(\lambda)$ and $Tr(\Pi_{k'}\phi_k^{\otimes t}) \leq \mathsf{negl}(\lambda)$ for all k and k' ($k \neq k'$).*

Proof. First, we show the if part. Assume that there exists a POVM measurement $\{\Pi_k\}_{k\in\{0,1\}^\kappa}$ and a polynomial t such that $\mathrm{Tr}(\Pi_k\phi_k^{\otimes t}) \geq 1-\mathsf{negl}(\lambda)$ and $\mathrm{Tr}(\Pi_{k'}\phi_k^{\otimes t}) \leq \mathsf{negl}(\lambda)$ for all k and k' ($k \neq k'$). Then,

$$\begin{aligned}\frac{t}{2}\|\phi_k - \phi_{k'}\|_1 &\geq \frac{1}{2}\|\phi_k^{\otimes t} - \phi_{k'}^{\otimes t}\|_1 \geq \mathrm{Tr}\big(\Pi_k\phi_k^{\otimes t}\big) - \mathrm{Tr}\big(\Pi_k\phi_{k'}^{\otimes t}\big) \geq 1 - \mathsf{negl}(\lambda) - \mathsf{negl}(\lambda)\\ &= 1 - \mathsf{negl}(\lambda),\end{aligned}$$

which means $\frac{1}{2}\|\phi_k - \phi_{k'}\|_1 \geq \frac{1}{t} - \mathsf{negl}(\lambda) \geq \frac{1}{2t}$.

Next, we show the only if part. Assume that the statistical invertibility is satisfied. Then, there exists a polynomial p such that $\frac{1}{2}\|\phi_k - \phi_{k'}\|_1 \geq \frac{1}{p}$ for all k and k' ($k \neq k'$). Let $t := 12p\kappa$. Then,

$$\frac{1}{2}\|\phi_k^{\otimes t} - \phi_{k'}^{\otimes t}\|_1 \geq 1 - e^{-t\frac{\|\phi_k - \phi_{k'}\|_1}{4}} \geq 1 - e^{-6\kappa} \geq 1 - 2^{-6\kappa},$$

which means $F(\phi_k^{\otimes t}, \phi_{k'}^{\otimes t}) \leq 2^{-6\kappa+1}$. From Theorem 40 below,

$$\max_k(1 - \mathrm{Tr}(\mu_k\phi_k^{\otimes t})) \leq \sum_{k\neq k'}\sqrt{F(\phi_k^{\otimes t}, \phi_{k'}^{\otimes t})} \leq 2^{-3\kappa+1}(2^{2\kappa} - 2^\kappa) \leq 2^{-\kappa+1},$$

which means $\mathrm{Tr}(\mu_k\phi_k^{\otimes t}) \geq 1 - 2^{-\kappa+1}$ and $\mathrm{Tr}(\mu_{k'}\phi_k^{\otimes t}) \leq 2^{-\kappa+1}$ for any k and k' ($k' \neq k$). ◀

▶ **Theorem 40** ([17]). *Let $\{\rho_i\}_i$ be a set of states. Define the POVM measurement $\{\mu_i\}_i$ with $\mu_i := \Sigma^{-1/2}\rho_i\Sigma^{-1/2}$, where $\Sigma := \sum_i \rho_i$, and the inverse is taken on the support of Σ. Then, $\max_i(1 - Tr(\mu_i\rho_i)) \leq \sum_{i\neq j}\sqrt{F(\rho_i, \rho_j)}$.*

7.2 Equivalence of SV-SI-OWSGs and EFI Pairs

▶ **Theorem 41.** *SV-SI-OWSGs exist if and only if EFI pairs exist.*

This Theorem is shown by combining the following two theorems.

▶ **Theorem 42.** *If EFI pairs exist then SV-SI-OWSGs exist.*

▶ **Theorem 43.** *If SV-SI-OWSGs exist then EFI pairs exist.*

Proof of Theorem 42. We show that if EFI pairs exist then SV-SI-OWSGs exist. Let $\mathsf{EFI.StateGen}(1^\lambda, b) \to \rho_b$ be an EFI pair. As is explained in Remark 2, we can assume without loss of generality that $\frac{1}{2}\|\rho_0 - \rho_1\|_1 \geq 1 - \mathsf{negl}(\lambda)$, which means $F(\rho_0, \rho_1) \leq \mathsf{negl}(\lambda)$. From the EFI pair, we construct a SV-SI-OWSG as follows.

- $\mathsf{KeyGen}(1^\lambda) \to k$: Choose $k \leftarrow \{0,1\}^\kappa$, and output k.
- $\mathsf{StateGen}(k) \to \phi_k$: Run $\mathsf{EFI.StateGen}(1^\lambda, k_i) \to \rho_{k_i}$ for each $i \in [\kappa]$. Output $\phi_k := \bigotimes_{i=1}^{\kappa} \rho_{k_i}$.

The statistical invertibility is easily shown as follows. If $k \neq k'$, there exists a $j \in [\kappa]$ such that $k_j \neq k'_j$. Then,

$$F(\phi_k, \phi_{k'}) \quad = \quad \prod_{i=1}^{\kappa} F(\rho_{k_i}, \rho_{k'_i}) \leq F(\rho_{k_j}, \rho_{k'_j}) \leq \mathsf{negl}(\lambda),$$

which means $\frac{1}{2}\|\phi_k - \phi_{k'}\|_1 \geq 1 - \mathsf{negl}(\lambda)$. This shows the statistical invertibility.

Let us next show the computational non-invertibility. From the standard hybrid argument, and the computational indistinguishability of ρ_0 and ρ_1, we have

$$\left| \frac{1}{2^\kappa} \sum_{k\in\{0,1\}^\kappa} \Pr\left[k \leftarrow \mathcal{A}(\phi_k^{\otimes t})\right] - \frac{1}{2^\kappa} \sum_{k\in\{0,1\}^\kappa} \Pr\left[k \leftarrow \mathcal{A}(\phi_{0^\kappa}^{\otimes t})\right] \right| \leq \mathsf{negl}(\lambda) \tag{3}$$

for any QPT adversary $\mathcal{A}$ and any polynomial t. (It will be shown later.) Hence

$$\begin{aligned} &\Pr\left[k \leftarrow \mathcal{A}(\phi_k^{\otimes t}) : k \leftarrow \mathsf{KeyGen}(1^\lambda), \phi_k \leftarrow \mathsf{StateGen}(k)\right] \\ =& \frac{1}{2^\kappa} \sum_{k\in\{0,1\}^\kappa} \Pr\left[k \leftarrow \mathcal{A}(\phi_k^{\otimes t})\right] \leq \frac{1}{2^\kappa} \sum_{k\in\{0,1\}^\kappa} \Pr\left[k \leftarrow \mathcal{A}(\phi_{0^\kappa}^{\otimes t})\right] + \mathsf{negl}(\lambda) = \frac{1}{2^\kappa} + \mathsf{negl}(\lambda), \end{aligned}$$

which shows the computational non-invertibility.

Let us show Eq. (3). For each $z \in \{0,1\}^{\kappa t}$, define $\Phi_z := \bigotimes_{i=1}^{\kappa t} \rho_{z_i}$. Let $z, z' \in \{0,1\}^{\kappa t}$ be two bit strings such that, for a single $j \in [\kappa t]$, $z_j = 0$, $z'_j = 1$, and $z_i = z'_i$ for all $i \neq j$. (In other words, z and z' are the same except for the jth bit.) Then, we can show that

$$\left| \frac{1}{2^\kappa} \sum_{k\in\{0,1\}^\kappa} \Pr[k \leftarrow \mathcal{A}(\Phi_z)] - \frac{1}{2^\kappa} \sum_{k\in\{0,1\}^\kappa} \Pr[k \leftarrow \mathcal{A}(\Phi_{z'})] \right| \leq \mathsf{negl}(\lambda) \tag{4}$$

for any QPT adversary $\mathcal{A}$. In fact, assume that

$$\left| \frac{1}{2^\kappa} \sum_{k} \Pr[k \leftarrow \mathcal{A}(\Phi_z)] - \frac{1}{2^\kappa} \sum_{k} \Pr[k \leftarrow \mathcal{A}(\Phi_{z'})] \right| \geq \frac{1}{\mathrm{poly}(\lambda)}$$

for a QPT adversary $\mathcal{A}$. Then, from this $\mathcal{A}$, we can construct a QPT adversary $\mathcal{B}$ that breaks the security of the EFI pair as follows: On input ρ_b, choose $k \leftarrow \{0,1\}^\kappa$, and run $k' \leftarrow \mathcal{A}((\bigotimes_{i=1}^{j-1} \rho_{z_i}) \otimes \rho_b \otimes (\bigotimes_{i=j+1}^{\kappa t} \rho_{z_i}))$. If $k' = k$, output $b' = 1$. If $k' \neq k$, output $b' = 0$. Because

$$\Pr[b' = 1|b = 0] = \frac{1}{2^\kappa} \sum_k \Pr[k \leftarrow \mathcal{A}(\Phi_z)], \quad \Pr[b' = 1|b = 1] = \frac{1}{2^\kappa} \sum_k \Pr[k \leftarrow \mathcal{A}(\Phi_{z'})],$$

we have $|\Pr[b' = 1|b = 0] - \Pr[b' = 1|b = 1]| \geq \frac{1}{\mathrm{poly}(\lambda)}$, which means that the $\mathcal{B}$ breaks the security of the EFI pair. From the standard hybrid argument and Eq. (4), we have Eq. (3). ◀

Proof of Theorem 43. We show that if SV-SI-OWSGs exist then EFI pairs exist. Let $(\mathsf{OWSG.KeyGen}, \mathsf{OWSG.StateGen})$ be a SV-SI-OWSG. Without loss of generality, we can assume that $\mathsf{OWSG.KeyGen}$ is the following algorithm: first apply a QPT unitary U on $|0...0\rangle$ to generate $U|0...0\rangle = \sum_k \sqrt{\Pr[k \leftarrow \mathsf{OWSG.KeyGen}(1^\lambda)]}|k\rangle|\mu_k\rangle$, and trace out the second register, where $\{|\mu_k\rangle\}_k$ are some normalized states. Moreover, without loss of generality, we can also assume that $\mathsf{OWSG.StateGen}$ is the following algorithm: first apply a QPT unitary V_k that depends on k on $|0...0\rangle$ to generate $V_k|0...0\rangle = |\psi_k\rangle_{\mathbf{A},\mathbf{B}}$, and trace out the register $\mathbf{A}$.

From the SV-SI-OWSG, we want to construct an EFI pair. For that goal, we construct a statistically-hiding and computationally-binding canonical quantum bit commitment scheme from SV-SI-OWSG. Due to Theorem 8 (the equivalence between different flavors of commitments), we then have a statistically-binding and computationally-hiding canonical quantum bit commitment scheme, which is equivalent to an EFI pair. From the SV-SI-OWSG, we construct a statistically-hiding and computationally-binding canonical quantum bit commitment scheme $\{Q_0, Q_1\}$ as follows.

$$\begin{aligned} Q_0|0\rangle_{\mathbf{C},\mathbf{R}} &\coloneqq \sum_k \sqrt{\Pr[k]}(|k\rangle|\mu_k\rangle)_{\mathbf{C}_1}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}, \\ Q_1|0\rangle_{\mathbf{C},\mathbf{R}} &\coloneqq \sum_k \sqrt{\Pr[k]}(|k\rangle|\mu_k\rangle)_{\mathbf{C}_1}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|k\rangle_{\mathbf{R}_3}, \end{aligned}$$

where $\Pr[k] \coloneqq \Pr\big[k \leftarrow \mathsf{OWSG.KeyGen}(1^\lambda)\big]$, $\mathbf{C}_2$ is the combination of all "$\mathbf{A}$ registers" of $|\psi_k\rangle$, $\mathbf{R}_2$ is the combination of all "$\mathbf{B}$ registers" of $|\psi_k\rangle$, $\mathbf{C} \coloneqq (\mathbf{C}_1, \mathbf{C}_2)$ and $\mathbf{R} \coloneqq (\mathbf{R}_2, \mathbf{R}_3)$. Moreover, t is a polynomial specified later. It is clear that such $\{Q_0, Q_1\}$ is implemented in QPT in a natural way.

Let us first show the computational binding of $\{Q_0, Q_1\}$. Assume that it is not computationally binding. Then, there exists a QPT unitary U, an ancilla state $|\tau\rangle$, and a polynomial p such that $\|(\langle 0|Q_1^\dagger)_{\mathbf{C},\mathbf{R}}U_{\mathbf{R},\mathbf{Z}}(Q_0|0\rangle_{\mathbf{C},\mathbf{R}} \otimes |\tau\rangle_{\mathbf{Z}})\| \geq \frac{1}{p}$. Then,

$$\begin{aligned} \frac{1}{p^2} &\leq \|(\langle 0|Q_1^\dagger)_{\mathbf{C},\mathbf{R}}U_{\mathbf{R},\mathbf{Z}}(Q_0|0\rangle_{\mathbf{C},\mathbf{R}} \otimes |\tau\rangle_{\mathbf{Z}})\|^2 \\ &= \Big\|\Big(\sum_{k'} \sqrt{\Pr[k']}\langle k', \mu_{k'}|_{\mathbf{C}_1}\langle\psi_{k'}|^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}\langle k'|_{\mathbf{R}_3}\Big) \\ &\quad\times\Big(\sum_k \sqrt{\Pr[k]}|k, \mu_k\rangle_{\mathbf{C}_1}U_{\mathbf{R},\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}|\tau\rangle_{\mathbf{Z}}\Big)\Big\|^2 \\ &= \Big\|\sum_k \Pr[k]\langle\psi_k|^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}\langle k|_{\mathbf{R}_3}U_{\mathbf{R},\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}|\tau\rangle_{\mathbf{Z}}\Big\|^2 \\ &\leq \Big(\sum_k \Pr[k]\Big\|\langle\psi_k|^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}\langle k|_{\mathbf{R}_3}U_{\mathbf{R},\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}|\tau\rangle_{\mathbf{Z}}\Big\|\Big)^2 \\ &\leq \sum_k \Pr[k]\Big\|\langle\psi_k|^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}\langle k|_{\mathbf{R}_3}U_{\mathbf{R},\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}|\tau\rangle_{\mathbf{Z}}\Big\|^2 \\ &\leq \sum_k \Pr[k]\Big\|\langle k|_{\mathbf{R}_3}U_{\mathbf{R},\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}|\tau\rangle_{\mathbf{Z}}\Big\|^2. \end{aligned} \tag{5}$$

In the third inequality, we have used Jensen's inequality.[12] From this U, we construct a QPT adversary $\mathcal{B}$ that breaks the computational non-invertibility of the SV-SI-OWSG as follows: On input the $\mathbf{R}_2$ register of $|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}$, apply $U_{\mathbf{R},\mathbf{Z}}$ on $|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0\rangle_{\mathbf{R}_3}|\tau\rangle_{\mathbf{Z}}$, and

[12] For a real convex function f, $f(\sum_i p_i x_i) \leq \sum_i p_i f(x_i)$.

measure the $\mathbf{R}_3$ register in the computational basis. Output the result. Then, the probability that $\mathcal{B}$ correctly outputs k is equal to Eq. (5). Therefore, $\mathcal{B}$ breaks the computational non-invertibility of the SV-SI-OWSG.

Let us next show the statistical hiding of $\{Q_0, Q_1\}$. In the following, we construct a (not necessarily QPT) unitary $W_{\mathbf{R},\mathbf{Z}}$ such that

$$\|W_{\mathbf{R},\mathbf{Z}}Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}} - Q_1|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}\|_1 \quad \le \quad \mathsf{negl}(\lambda). \tag{6}$$

Then, we have

$$\begin{aligned}
&\|\mathrm{Tr}_{\mathbf{R}}(Q_0|0\rangle_{\mathbf{C},\mathbf{R}}) - \mathrm{Tr}_{\mathbf{R}}(Q_1|0\rangle_{\mathbf{C},\mathbf{R}})\|_1 = \|\mathrm{Tr}_{\mathbf{R},\mathbf{Z}}(Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}) - \mathrm{Tr}_{\mathbf{R},\mathbf{Z}}(Q_1|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}})\|_1\\
= \;&\|\mathrm{Tr}_{\mathbf{R},\mathbf{Z}}(W_{\mathbf{R},\mathbf{Z}}Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}) - \mathrm{Tr}_{\mathbf{R},\mathbf{Z}}(Q_1|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}})\|_1\\
\le \;&\|W_{\mathbf{R},\mathbf{Z}}Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}} - Q_1|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}\|_1 \le \mathsf{negl}(\lambda),
\end{aligned}$$

which shows the statistical hiding of $\{Q_0, Q_1\}$.

Now we explain how to construct $W_{\mathbf{R},\mathbf{Z}}$. From Lemma 39, there exists a (not necessarily QPT) POVM measurement $\{\Pi_k\}_k$ and a polynomial t such that $\mathrm{Tr}(\Pi_k\phi_k^{\otimes t}) \ge 1 - \mathsf{negl}(\lambda)$ and $\mathrm{Tr}(\Pi_{k'}\phi_k^{\otimes t}) \le \mathsf{negl}(\lambda)$ for all k and k' $(k \ne k')$. Let $U_{\mathbf{R}_2,\mathbf{Z}}$ be a unitary operator that implements the POVM measurement $\{\Pi_k\}_k$ in the following way

$$U_{\mathbf{R}_2,\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0...0\rangle_{\mathbf{Z}} \quad = \quad \sqrt{1-\epsilon_k}|k\rangle|junk_k\rangle + \sum_{k':k'\ne k}\sqrt{\epsilon_{k'}}|k'\rangle|junk_{k'}\rangle,$$

where $\mathbf{Z}$ is the ancilla register, $\{\epsilon_i\}_i$ are real numbers such that $1-\epsilon_k \ge 1-\mathsf{negl}(\lambda)$ and $\epsilon_{k'} \le \mathsf{negl}(\lambda)$ for all $k' \ne k$, and $\{|junk_i\rangle\}_i$ are "junk" states that are normalized. Measuring the first register of the state realizes the POVM. Let $V_{\mathbf{R},\mathbf{Z}}$ be the following unitary:[13]

1. Apply $U_{\mathbf{R}_2,\mathbf{Z}}$ on $|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0...0\rangle_{\mathbf{Z}}|0\rangle_{\mathbf{R}_3}$:
$$U_{\mathbf{R}_2,\mathbf{Z}}|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0...0\rangle_{\mathbf{Z}}|0\rangle_{\mathbf{R}_3} \quad = \quad \Big[\sqrt{1-\epsilon_k}|k\rangle|junk_k\rangle + \sum_{k':k'\ne k}\sqrt{\epsilon_{k'}}|k'\rangle|junk_{k'}\rangle\Big]|0\rangle_{\mathbf{R}_3}.$$
2. Copy the content of the first register to the register $\mathbf{R}_3$:
$$\sqrt{1-\epsilon_k}|k\rangle|junk_k\rangle|k\rangle_{\mathbf{R}_3} + \sum_{k':k'\ne k}\sqrt{\epsilon_{k'}}|k'\rangle|junk_{k'}\rangle|k'\rangle_{\mathbf{R}_3}.$$

Define $W_{\mathbf{R},\mathbf{Z}} := U^{\dagger}_{\mathbf{R}_2,\mathbf{Z}}V_{\mathbf{R},\mathbf{Z}}$.

Let us show that thus constructed $W_{\mathbf{R},\mathbf{Z}}$ satisfies Eq. (6).

$$\begin{aligned}
&\Big((\langle 0|Q_1^{\dagger})_{\mathbf{C},\mathbf{R}}\langle 0|_{\mathbf{Z}}\Big)\Big(W_{\mathbf{R},\mathbf{Z}}Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}\Big)\\
&= \Big((\langle 0|Q_1^{\dagger})_{\mathbf{C},\mathbf{R}}\langle 0|_{\mathbf{Z}}\Big)\Big(U^{\dagger}_{\mathbf{R}_2,\mathbf{Z}}V_{\mathbf{R},\mathbf{Z}}Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}\Big)\\
&= \Big((\langle 0|Q_1^{\dagger})_{\mathbf{C},\mathbf{R}}\langle 0|_{\mathbf{Z}}U^{\dagger}_{\mathbf{R}_2,\mathbf{Z}}\Big)\Big(V_{\mathbf{R},\mathbf{Z}}Q_0|0\rangle_{\mathbf{C},\mathbf{R}}|0\rangle_{\mathbf{Z}}\Big)\\
&= \Big(\sum_k\sqrt{\Pr[k]}(\langle k|\langle\mu_k|)_{\mathbf{C}_1}\Big[\sqrt{1-\epsilon_k}\langle k|\langle junk_k|\langle k|_{\mathbf{R}_3} + \sum_{k'\ne k}\sqrt{\epsilon_{k'}}\langle k'|\langle junk_{k'}|\langle k|_{\mathbf{R}_3}\Big]\Big)\\
&\times\Big(\sum_k\sqrt{\Pr[k]}(|k\rangle|\mu_k\rangle)_{\mathbf{C}_1}\Big[\sqrt{1-\epsilon_k}|k\rangle|junk_k\rangle|k\rangle_{\mathbf{R}_3} + \sum_{k'\ne k}\sqrt{\epsilon_{k'}}|k'\rangle|junk_{k'}\rangle|k'\rangle_{\mathbf{R}_3}\Big]\Big)\\
&= \sum_k\Pr[k](1-\epsilon_k) \ge 1-\mathsf{negl}(\lambda).
\end{aligned}$$
◀

[13] For simplicity, we define $V_{\mathbf{R},\mathbf{Z}}$ by explaining how it acts on $|\psi_k\rangle^{\otimes t}_{\mathbf{C}_2,\mathbf{R}_2}|0...0\rangle_{\mathbf{Z}}|0\rangle_{\mathbf{R}_3}$, but it is clear from the explanation how $V_{\mathbf{R},\mathbf{Z}}$ is defined.

References

1 Scott Aaronson. Shadow tomography of quantum states. *SIAM J. Comput.*, 49(5):STOC18–368, 2019.

2 Scott Aaronson and Paul Christiano. Quantum money from hidden subspaces. In Howard J. Karloff and Toniann Pitassi, editors, *44th ACM STOC*, pages 41–60. ACM Press, May 2012. doi:10.1145/2213977.2213983.

3 Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. Cryptology ePrint Archive, Paper 2021/1663, 2021. URL: https://eprint.iacr.org/2021/1663.

4 Zvika Brakerski, Ran Canetti, and Luowen Qian. On the computational hardness needed for quantum cryptography. Cryptology ePrint Archive, Paper 2022/1181, 2022. URL: https://eprint.iacr.org/2022/1181.

5 Paul Dumais, Dominic Mayers, and Louis Salvail. Perfectly concealing quantum bit commitment from any quantum one-way permutation. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 300–315. Springer, Heidelberg, May 2000. doi:10.1007/3-540-45539-6_21.

6 Junbin Fang, Dominique Unruh, Jun Yan, and Dehua Zhou. How to base security on the perfect/statistical binding property of quantum bit commitment? Cryptology ePrint Archive, Report 2020/621, 2020. URL: https://ia.cr/2020/621.

7 Rosario Gennaro, Yael Gertner, Jonathan Katz, and Luca Trevisan. Bounds on the efficiency of generic cryptographic constructions, 2005. doi:10.1137/S0097539704443276.

8 Oded Goldreich. A note on computational indistinguishability. Information Processing Letters 34.6 (1990), pp.277–281., 1990. doi:10.1016/0020-0190(90)90010-U.

9 Oded Goldreich. *The Foundations of Cryptography - Volume 1: Basic Techniques*. Cambridge University Press, 2001. doi:10.1017/CBO9780511546891.

10 Minki Hhan, Tomoyuki Morimae, and Takashi Yamakawa. From the hardness of detecting superpositions to cryptography: Quantum public key encryption and commitments. *arXiv*, 2022. arXiv:2210.05978.

11 Russell Impagliazzo. A personal view of average-case complexity. In *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*, pages 134–147. IEEE Computer Society, 1995. doi:10.1109/SCT.1995.514853.

12 Gene Itkis, Emily Shen, Mayank Varia, David Wilson, and Arkady Yerukhimovich. Bounded-collusion attribute-based encryption from minimal assumptions. In Serge Fehr, editor, *PKC 2017, Part II*, volume 10175 of *LNCS*, pages 67–87. Springer, Heidelberg, March 2017. doi:10.1007/978-3-662-54388-7_3.

13 Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 126–152. Springer, Heidelberg, August 2018. doi:10.1007/978-3-319-96878-0_5.

14 W. Kretschmer. Quantum pseudorandomness and classical complexity. *TQC 2021*, 2021. doi:10.4230/LIPICS.TQC.2021.2.

15 William Kretschmer, Luowen Qian, Makrand Sinha, and Avishay Tal. Quantum cryptography in algorithmica. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 1589–1602, New York, NY, USA, 2023. Association for Computing Machinery. doi:10.1145/3564246.3585225.

16 Ching-Yi Lai and Kai-Min Chung. Quantum encryption and generalized quantum shannon impossibility. *Designs, Codes and Cryptography volume 87, pages 1961–1972 (2019)*, 2019. doi:10.1007/s10623-018-00597-3.

17 Ashley Montanaro. Pretty simple bounds on quantum state discrimination. *arXiv*, 2019. doi:10.48550/arXiv.1908.08312.

18 Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. Cryptology ePrint Archive, Paper 2021/1691, 2021. URL: https://eprint.iacr.org/2021/1691.

19 Dominique Unruh. Computationally binding quantum commitments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 497–527. Springer, Heidelberg, May 2016. doi:10.1007/978-3-662-49896-5_18.

20 Jun Yan. General properties of quantum bit commitments. Cryptology ePrint Archive, Paper 2020/1488, 2020. URL: https://eprint.iacr.org/2020/1488.

21 Jun Yan. Quantum computationally predicate-binding commitments with application in quantum zero-knowledge arguments for NP. In *ASIACRYPT 2021, Part I*, LNCS, pages 575–605. Springer, Heidelberg, December 2021. doi:10.1007/978-3-030-92062-3_20.

22 Jun Yan, Jian Weng, Dongdai Lin, and Yujuan Quan. Quantum bit commmitment with application in quantum zero-knowledge proof (extended abstract). In Khaled M. Elbassioni and Kazuhisa Makino, editors, *ISAAC 2015*, volume 9472 of *Lecture Notes in Computer Science*, pages 555–565. Springer, 2015. doi:10.1007/978-3-662-48971-0_47.

23 Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd FOCS*, pages 80–91. IEEE Computer Society Press, November 1982. doi:10.1109/SFCS.1982.45.

Revocable Quantum Digital Signatures

Tomoyuki Morimae
Yukawa Institute for Theoretical Physics, Kyoto University, Japan

Alexander Poremba
Computing and Mathematical Sciences, Caltech, Pasadena, CA, USA
CSAIL and Department of Mathematics, MIT, Cambridge, MA, USA

Takashi Yamakawa
NTT Social Informatics Laboratories, Tokyo, Japan
NTT Research Center for Theoretical Quantum Information, Atsugi, Japan

Abstract

We study digital signatures with *revocation capabilities* and show two results. First, we define and construct digital signatures with revocable signing keys from the LWE assumption. In this primitive, the signing key is a quantum state which enables a user to sign many messages and yet, the quantum key is also *revocable*, i.e., it can be collapsed into a classical certificate which can later be verified. Once the key is successfully revoked, we require that the initial recipient of the key loses the ability to sign. We construct digital signatures with revocable signing keys from a newly introduced primitive which we call *two-tier one-shot signatures*, which may be of independent interest. This is a variant of one-shot signatures, where the verification of a signature for the message "0" is done publicly, whereas the verification for the message "1" is done in private. We give a construction of two-tier one-shot signatures from the LWE assumption. As a complementary result, we also construct digital signatures with *quantum* revocation from group actions, where the quantum signing key is simply "returned" and then verified as part of revocation.

Second, we define and construct digital signatures with revocable signatures from OWFs. In this primitive, the signer can produce quantum signatures which can later be revoked. Here, the security property requires that, once revocation is successful, the initial recipient of the signature loses the ability to find accepting inputs to the signature verification algorithm. We construct this primitive using a newly introduced *two-tier* variant of tokenized signatures. For the construction, we show a new lemma which we call the adaptive hardcore bit property for OWFs, which may enable further applications.

2012 ACM Subject Classification Theory of computation → Cryptographic protocols

Keywords and phrases Quantum cryptography, digital signatures, revocable cryptography

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.5

Related Version *Full Version*: `https://eprint.iacr.org/2023/1937`

Funding *Tomoyuki Morimae*: JST CREST JPMJCR23I3, JST Moonshot R&D JPMJMS2061-5-1-1, JST FOREST, MEXT QLEAP, the Grant-in-Aid for Scientific Research (B) No. JP19H04066, the Grant-in Aid for Transformative Research Areas (A) 21H05183, and the Grant-in-Aid for Scientific Research (A) No. 22H00522.
Alexander Poremba: National Science Foundation (NSF) under Grant No. CCF-1729369.
Takashi Yamakawa: Supported by the NTT Corporation.

1 Introduction

1.1 Background

The exotic nature of quantum physics, such as quantum superposition, no-cloning, entanglement, and uncertainty relations, enables many new cryptographic applications which are impossible in a classical world. These include quantum money [43], copy-protection [1, 2],

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 5; pp. 5:1–5:24
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

secure software leasing [5], unclonable encryption [20, 15], certified deletion [14], and more. Here, a common approach is to encode information into a quantum state which prevents it from being copied by the no-cloning principle. [42, 14, 23, 24, 6, 11, 9, 30, 35, 8, 7]

Following this line of research, Ananth, Poremba, and Vaikuntanathan [6] and Agrawal, Kitagawa, Nishimaki, Yamada, and Yamakawa [3] concurrently introduced the concept of key-revocable public key encryption (PKE),[1] which realizes the following functionality: a decryption capability is delegated to a user in the form of a quantum decryption key in such a way that, once the key is returned, the user loses the ability to decrypt. They constructed key-revocable PKE schemes based on standard assumptions, namely quantum hardness of the learning with errors problem (LWE assumption) [6] or even the mere existence of any PKE scheme [3]. They also extended the idea of revocable cryptography to pseudorandom functions [6] and encryption with advanced functionality such as attribute-based encryption and functional encryption [3]. However, neither of these works extended the idea to *digital signatures* despite their great importance in cryptography. This state of affairs raises the following question:

Is it possible to construct digital signature schemes with revocation capabilities?

The delegation of privileges is of central importance in cryptography, and the task of revoking privileges in the context of digital signatures and certificates, in particular, remains a fundamental challenge for cryptography [41, 39]. One simple solution is to use a limited-time delegatable signature scheme, where a certified signing key is generated together with an expiration date. Note that this requires that the expiration date is known ahead of time and that the clocks be synchronized. Moreover, issuing new keys (for example, each day) could potentially also be costly. Quantum digital signature schemes with revocation capabilities could potentially resolve these difficulties by leveraging the power of quantum information.

To illustrate the use of *revocable* digital signature schemes, consider the following scenarios. Suppose that an employee at a company, say Alice, takes a temporary leave of absence and wishes to authorize her colleague, say Bob, to sign a few important documents on her behalf. One thing Alice can do is to simply use a (classical) digital signature scheme and to share her signing keys with Bob. While this naïve approach would certainly allow Bob to produce valid signatures while Alice is gone, it also means that Bob continues to have access to the signing keys – long after Alice's return. This is because the signing key of a digital signature scheme is *classical*, and hence it can be copied at will. In particular, a malicious Bob could secretly sell Alice's signing key to a third party for a profit. A digital signature scheme with *revocable signing keys* can remedy this situation as it enables Alice to certify that Bob has lost access to the signing key once and for all.

As a second example, consider the following scenario. Suppose that a company or a governmental organization wishes to grant a new employee certain access privileges throughout their employment; for example to various buildings or office spaces. One solution is to use an *electronic* ID card through a mobile device, where a digital signature is used for identity management. Naturally, one would like to ensure that, once the employee's contract is terminated, their ID card is disabled in the system and no longer allows for further unauthorized access. However, if the signature corresponding to the employee's ID is a digital object, it is conceivable that the owner of the card manages to retain their ID card even after it is disabled. This threat especially concerns scenarios in which the verification of an ID

[1] Agrawal et al. [3] call it PKE with secure key leasing.

card is performed by a device which is not connected to the internet, or simply not updated frequently enough. A digital signature scheme with revocable signatures can remedy this situation as it enables *revocable quantum* ID cards; in particular, it allows one to certify that the initial access privileges have been revoked once and for all.

1.2 Our Results

In this paper, we show the following two results on revocable digital signatures.

Revocable signing keys. First, we define digital signatures with revocable *signing keys* (DSR-Key). In this primitive, a signing key is encoded in the form of a quantum state which enables the recipient to sign many messages. However, once the key is successfully revoked from a user, they no longer have the ability to generate valid signatures. Here, we consider *classical revocation*, i.e., a classical certificate is issued once the user destroys the quantum signing key with an appropriate measurement. In addition, the verification of the revocation certificate takes place in private, which means that the verification requires a private key which should be kept secret. We construct DSR-Key based solely on the quantum hardness of the LWE problem [38]. We remark that our scheme is inherently *stateful*, i.e., whenever a user generates a new signature, the user must update the singing key for the next invocation of the signing algorithm. Indeed, we believe that digital signatures with revocable signing keys must be inherently stateful since a user must keep the quantum signing key as a "state" for generating multiple signatures. An undesirable feature of our scheme is that the signing key and signature sizes grow with the number of signatures to be generated.

As complementary result, we also consider DSR-Key with *quantum* revocation. In this primitive, not a classical deletion certificate but the quantum signing key itself is returned for the revocation. In the full version of the paper, we construct the primitive from group actions with the one-wayness property [26]. The existence of group actions with the one-wayness property is incomparable with the LWE assumption.

Revocable signatures. Second, we define digital signatures with revocable *signatures* (DSR-Sign). In this primitive, signatures are encoded as quantum states which can later be revoked. The security property guarantees that, once revocation is successful, the initial recipient of the signature loses the ability to pass the signature verification. We construct digital signatures with revocable signatures based on the existence of (quantum-secure) one-way functions (OWFs). In our scheme, the revocation is classical and private, i.e., a user can issue a classical certificate of revocation, which is verified by using a private key.

1.3 Comparison with Existing Works

To our knowledge, there is no prior work that studies digital signatures with quantum signatures. On the other hand, there are several existing works that study digital signatures with quantum signing keys. We review them and compare them with our DSR-Key.

- **Tokenized signatures [12, 17].** In a tokenized signature scheme, the signing key corresponds to a quantum state which can be used to generate a signature on at most one message. At first sight, the security notion seems to imply the desired security guarantee for DSR-Key, since a signature for a dummy message may serve as the classical deletion certificate for the signing key.[2] However, the problem is that tokenized signatures do not achieve the correctness for DSR-Key; namely, in tokenized signatures, a user who receives a quantum signing key can generate only a single signature, whereas in DSR-Key, we require that a user can generate arbitrarily many signatures before the signing key is revoked. Thus, tokenized signatures are not sufficient for achieving our goal. A similar problem exists for semi-quantum tokenized signatures [40] and one-shot signatures [4] as well.

- **Copy-protection for digital signatures [31] (a.k.a. single-signer signatures [4].[3])** In this primitive, a signing key corresponds to a quantum state which cannot be copied. More precisely, suppose that a user is given one copy of the signing key and tries to split it into two signing keys. The security property requires that at most one of these two signing keys is capable at generating a valid signature on a random message. Amos, Georgiou, Kiayias, and Zhandry [4] constructed such a signature scheme based on one-shot signatures. However, the only known construction of one-shot signatures is relative to classical oracles, and there is no known construction without oracles. Liu, Liu, Qian, and Zhandry [31] constructed it based on indistinguishability obfuscation (iO) and OWFs. Intuitively, copy-protection for digital signatures implies DSR-Key, because checking whether a returned signing key succeeds at generating valid signatures on random messages can serve a means of verification for revocation.[4] Compared with this approach, our construction has the advantage that it is based on the standard assumption (namely the LWE assumption), whereas they require the very strong assumption of iO or ideal oracles. On the other hand, a disadvantage of our construction is that revocation requires private information, whereas theirs have the potential for public revocation. Another disadvantage is that the size of the signing key (and signatures) grows with the number of signatures, whereas this is kept constant in [31] (but not in [4]).

1.4 Technical Overview

Here we give intuitive explanations of our constructions.

Construction of DSR-Key. Our first scheme, DSR-Key, is constructed using *two-tier one-shot signatures* (2-OSS), which is a new primitive which we introduce in this paper.[5] 2-OSS are variants of one-shot signatures [4] for single-bit messages. The main difference with regard to one-shot signatures is that there are two verification algorithms, and a signature for the message “0” is verified by a public verification algorithm, whereas a signature for the massage “1” is verified by a *private* verification algorithm. We believe that the notion of 2-OSS may be of independent interest. Our construction of 2-OSS is conceptually similar to the construction of two-tier quantum lightning in [29], and can be based solely on the LWE assumption.

[2] Note, however, that tokenized signatures offer public verification of signatures, whereas certifying revocation in our DSR-Key scheme takes place in private.

[3] Technically speaking, [31] and [4] require slightly different security definitions, but high level ideas are the same.

[4] While this sounds plausible, there is a subtlety regarding the security definitions. Indeed, we believe that the security of copy-protection for digital signatures [31] or single-signer signatures [4] does not readily imply our security definition in Definition 4, though they do seem to imply some weaker but reasonable variants of security. See also Remark 5.

[5] The term “two-tier” is taken from [29] where they define two-tier quantum lightning, which is a similar variant of quantum lightning [44].

From 2-OSS, we then go on to construct $\mathsf{DSR\text{-}Key}$. We first construct $\mathsf{DSR\text{-}Key}$ for single-bit messages from 2-OSS as follows.[6] The signing key sigk of $\mathsf{DSR\text{-}Key}$ consists of a pair $(\mathsf{sigk}_0, \mathsf{sigk}_1)$ of signing keys of a 2-OSS scheme. To sign a single-bit message $m \in \{0,1\}$, the message "0" is signed with the signing algorithm of a 2-OSS scheme using the signing key sigk_m. Because the signature on m corresponds to a particular signature of "0" with respect to the 2-OSS scheme, it can be verified with the public verification algorithm of 2-OSS. To delete the signing key, the message "1" is signed with the signing algorithm of 2-OSS by using the signing key. The signature for the message "1" corresponds to the revocation certificate, and it can be verified using the private verification algorithm of 2-OSS.

Our aforementioned construction readily implies a *one-time version* of a $\mathsf{DSR\text{-}Key}$ scheme, namely, the correctness and security hold when the signing is used only once. We then upgrade it to the many-time version by using a similar chain-based construction of single-signer signatures from one-shot signatures as in [4]. That is, it works as follows. The signing key and verification key of the many-time scheme are those of the one-time scheme, respectively. We denote them by $(\mathsf{ot.sigk}_0, \mathsf{ot.vk}_0)$. When signing on the first message m_1, the signer first generates a new key pair $(\mathsf{ot.sigk}_1, \mathsf{ot.vk}_1)$ of the one-time scheme, uses $\mathsf{ot.sigk}_0$ to sign on the concatenation $m_1 \| \mathsf{ot.vk}_1$ of the message and the newly generated verification key to generate a signature $\mathsf{ot}.\sigma_1$ of the one-time scheme. Then it outputs $(m_1, \mathsf{ot.vk}_1, \mathsf{ot}.\sigma_1)$ as a signature of the many-time scheme.[7] Similarly, when signing on the k-th message m_k for $k \geq 2$, the signer generates a new key pair $(\mathsf{ot.sigk}_k, \mathsf{ot.vk}_k)$ and uses $\mathsf{ot.sigk}_{k-1}$ to sign on $m_k \| \mathsf{ot.vk}_k$ to generate a signature $\mathsf{ot}.\sigma_k$. Then the signature of the many-time scheme consists of $\{m_i, \mathsf{ot.vk}_i, \mathsf{ot}.\sigma_i\}_{i\in[k]}$. The verification algorithm of the many-time scheme verifies $\mathsf{ot}.\sigma_i$ for all $i \in [k]$ under the corresponding message and verification key, and accepts if all of these verification checks pass. To revoke a signing key, the signer generates revocation certificates for all of the signing keys of the one-time scheme which have previously been generated, and the verification of the revocation certificate simply verifies that all these revocation certificates are valid.[8] It is easy to reduce security of the above many-time scheme to that of the one-time scheme.

Construction of DSR-Sign. Our second scheme, $\mathsf{DSR\text{-}Sign}$, is constructed from what we call *two-tier tokenized signatures* (2-TS), which is a new primitive introduced in this paper. 2-TS are variants of tokenized signatures [12] for single-bit messages where two signature verification algorithms exist. One verification algorithm is used to verify signatures for the message "0", and it uses the public key. The other verification algorithm is used to verify signatures for the message "1", and it uses the *secret* key.

We construct 2-TS from OWFs by using a new lemma that we call the *adaptive hardcore bit property for OWFs*, inspired by a similar notion which was shown for a family of noisy trapdoor claw-free functions by Brakerski et al. [13]. We believe that our lemma may be of independent interest, and enable further applications down the line. The adaptive hardcore bit property for OWFs roughly states that given $|x_0\rangle + (-1)^c |x_1\rangle$ and $(f(x_0), f(x_1))$, no QPT adversary can output (x, d) such that $f(x) \in \{f(x_0), f(x_1)\}$ and $d \cdot (x_0 \oplus x_1) = c$, where f is a OWF, $x_0, x_1 \leftarrow \{0,1\}^\ell$, and $c \leftarrow \{0,1\}$.[9] The adaptive hardcore bit property for OWFs is shown by using a theorem which is implicit in a recent work [10].

[6] The scheme can be extended to the one for multi-bit messages by using the collision resistant hash functions.

[7] We include m_1 in the signature for notational convenience even though this is redundant.

[8] The ability to verify all previously generated signing keys (e.g., as part of a chain) may require secret *trapdoor information.*

[9] We actually need its amplified version, because in this case the adversary can win with probability 1/2 by measuring the state to get x_0 or x_1, and randomly choosing d.

From the adaptive hardcore bit property for OWFs, we construct 2-TS as follows: The quantum signing token is $|x_0\rangle + (-1)^c |x_1\rangle$ with random $x_0, x_1 \leftarrow \{0,1\}^\ell$ and $c \leftarrow \{0,1\}$.[10] The public key is $(f(x_0), f(x_1))$, where f is a OWF, and the secret key is (x_0, x_1, c). To sign the message "0", the token is measured in the computational basis to obtain either x_0 or x_1. To sign the message "1", the token is measured in the Hadamard basis to obtain a string d such that $d \cdot (x_0 \oplus x_1) = c$. The measurement result in the computational basis is then verified with the public key, whereas the measurement result in the Hadamard basis is verified with the secret key. Due to the adaptive hardcore bit property for OWFs (formally shown in Theorem 9), no QPT adversary can output both signatures at the same time.

Finally, we observe that DSR-Sign can be constructed from any 2-TS scheme by considering the quantum signature of DSR-Sign as a quantum signing token of 2-TS. To verify the quantum signature, we sign the message "0" by using the quantum token, and verify it. To delete the quantum signature, we sign the message "1" by using the quantum token. The verification of the revocation certificate requires one to check whether the deletion certificate is a valid signature for message "1" or not.

1.5 Related Works

We have already explained relations between our results and existing works on digital signatures with quantum signing keys. Here, we give a brief review on other related quantum cryptographic primitives.

Certified deletion and revocation. Unruh [42] first initiated the study of quantum revocable encryption. This allows the recipient of a quantum ciphertext to return the state, thereby losing all information about the encrypted message. Quantum encryption with certified deletion [23, 36, 8, 22, 7, 11], first introduced by Broadbent and Islam [14], enables the deletion of quantum ciphertexts, whereby a classical certificate is produced which can be verified. In particular, [8, 22, 24] study the certified everlasting security where the security is guaranteed even against unbounded adversary once a valid deletion certificate is issued. [30] and [10] recently showed a general conversion technique to convert the certified everlasting lemma by Bartusek and Kurana [8] for the private verification to the public one assuming only OWFs (or even weaker assumptions such as hard quantum planted problems for **NP** or the one-way states generators [33]).

The notion of certified deletion has also been used to revoke cryptographic keys [28, 3, 7, 6, 16]. Here, a key is delegated to a user in the form of a quantum state which can later be revoked. Once the key is destroyed and a valid certificate is issued, the functionality associated with the key is no longer available to the user.

Finally, we remark that the notion of revocation has also been considered in the context of more general programs. Ananth and La Placa [5] introduced the notion of secure software leasing. Here, the security guarantees that the functionality of a piece of quantum software is lost once it is returned and verified.

Copy-protection. Copy-protection, introduced by Aaronson [1], is a primitive which allows one to encode a functionality into a quantum state in such a way that it cannot be cloned. [2] showed that any unlearnable functionality can be copy-protected with a classical

[10] Again, we actually consider its amplified version so that the winning probability of the adversary is negligibly small.

oracle. [18] constructed copy-protection schemes for (multi-bit) point functions as well as compute-and-compare programs in the quantum random oracle model. [17] constructed unclonable decryption schemes from iO and compute-and-compare obfuscation for the class of unpredictable distributions, which were previously constructed with classical oracle in [19]. [17] also constructed a copy-protection scheme for pseudorandom functions assuming iO, OWFs, and compute-and-compare obfuscation for the class of unpredictable distributions. [31] constructed bounded collusion-resistant copy-protection for various functionalities (copy-protection of decryption, digital signatures and PRFs) with iO and LWE.

2 Preliminaries

2.1 Basic Notation

We use the standard notations of quantum computing and cryptography. We use λ as the security parameter. For any set S, $x \leftarrow S$ means that an element x is sampled uniformly at random from the set S. We write negl to mean a negligible function. PPT stands for (classical) probabilistic polynomial-time and QPT stands for quantum polynomial-time. For an algorithm A, $y \leftarrow A(x)$ means that the algorithm A outputs y on input x. For two bit strings x and y, $x\|y$ means the concatenation of them. For simplicity, we sometimes omit the normalization factor of a quantum state. (For example, we write $\frac{1}{\sqrt{2}}(|x_0\rangle + |x_1\rangle)$ just as $|x_0\rangle + |x_1\rangle$.) $I := |0\rangle\langle 0| + |1\rangle\langle 1|$ is the two-dimensional identity operator. For the notational simplicity, we sometimes write $I^{\otimes n}$ just as I when the dimension is clear from the context. For two density matrices ρ and σ, the trace distance is defined as

$$\|\rho - \sigma\|_{\mathrm{tr}} := \frac{1}{2}\|\rho - \sigma\|_1 = \frac{1}{2}\mathrm{Tr}\left[\sqrt{(\rho - \sigma)^2}\right],$$

where $\|\cdot\|_1$ is the trace norm. We also make use of the following result.

▶ **Theorem 1** (Holevo-Helstrom, [25, 21]). *Consider an experiment in which one of two quantum states, either ρ or σ, is sent to a distinguisher with probability $1/2$. Then, any measurement which seeks to discriminate between ρ and σ has success probability p_{succ} at most $p_{succ} \leq \frac{1}{2} + \frac{1}{2}\|\rho - \sigma\|_{\mathrm{tr}}$.*

3 Two-Tier One-Shot Signatures

In this section, we define two-tier one-shot signatures (2-OSS), and construct it from the LWE assumption [38]. Broadly speaking, this cryptographic primitive is a variant of one-shot signatures [4], where the verification of a signature for the message "0" is done publicly, whereas that for the message "1" is done only privately.

3.1 Definition

The formal definition of 2-OSS is as follows.

▶ **Definition 2** (Two-Tier One-Shot Signatures (2-OSS)). *A two-tier one-shot signature scheme is a set* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Ver}_0, \mathsf{Ver}_1)$ *of algorithms such that*

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{sk})$: *on input the security parameter λ, it outputs a classical parameter* pp *and a classical secret key* sk.
- $\mathsf{KeyGen}(\mathsf{pp}) \to (\mathsf{sigk}, \mathsf{vk})$: *on input* pp, *it outputs a quantum signing key* sigk *and a classical verification key* vk.

- $\mathsf{Sign}(\mathsf{sigk}, m) \to \sigma$: *on input* sigk *and a single-bit message* $m \in \{0,1\}$*, it outputs a classical signature* σ*.*
- $\mathsf{Ver}_0(\mathsf{pp}, \mathsf{vk}, \sigma) \to \top/\bot$: *on input* pp*,* vk*, and* σ*, it outputs* $\top/\bot$*.*
- $\mathsf{Ver}_1(\mathsf{pp}, \mathsf{sk}, \mathsf{vk}, \sigma) \to \top/\bot$: *on input* pp*,* sk*, and* σ*, it outputs* $\top/\bot$*.*

We require the following properties.

Correctness:

$$\Pr\left[\top \leftarrow \mathsf{Ver}_0(\mathsf{pp}, \mathsf{vk}, \sigma) : \begin{array}{r} (\mathsf{sk}, \mathsf{pp}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp}) \\ \sigma \leftarrow \mathsf{Sign}(\mathsf{sigk}, 0) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda) \tag{1}$$

and

$$\Pr\left[\top \leftarrow \mathsf{Ver}_1(\mathsf{pp}, \mathsf{sk}, \mathsf{vk}, \sigma) : \begin{array}{r} (\mathsf{sk}, \mathsf{pp}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp}) \\ \sigma \leftarrow \mathsf{Sign}(\mathsf{sigk}, 1) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \tag{2}$$

Security: *For any QPT adversary* $\mathcal{A}$*,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}_0(\mathsf{pp}, \mathsf{vk}, \sigma_0) \wedge \top \leftarrow \mathsf{Ver}_1(\mathsf{pp}, \mathsf{sk}, \mathsf{vk}, \sigma_1) : \begin{array}{c} (\mathsf{sk}, \mathsf{pp}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{vk}, \sigma_0, \sigma_1) \leftarrow \mathcal{A}(\mathsf{pp}) \end{array}\right] \leq \mathsf{negl}(\lambda). \tag{3}$$

3.2 Construction

We show that 2-OSS can be constructed from the LWE assumption [38]. Specifically, we make use of *noisy trapdoor claw-free function* (NTCF) families which allow us to generate quantum states that have a nice structure in both the computational basis, as well as the Hadamard basis. For a detailed definition of NTCF families, we refer to [13].

Our 2-OSS scheme is based on the two-tier quantum lightning scheme in [29] and leverages this structure to sign messages: to sign the message "0", we output a measurement outcome in the computational basis, whereas if we wish to sign "1", we output a measurement outcome in the Hadamard basis. Crucially, the so-called adaptive hardcore-bit property ensures that it is computationally difficult to produce the two outcomes simultaneously. In this context, we use the amplified adaptive hardcore bit property which was shown in [37, 29].

In the full version of the paper, we show the following result.

▶ **Theorem 3.** *Assuming the quantum hardness of the LWE problem, there exists two-tier one-shot signatures.*

4 Digital Signatures with Revocable Signing Keys

In this section, we define digital signatures with revocable signing keys (DSR-Key) and give its construction from 2-OSS.

4.1 Definition

Let us now present a formal definition of DSR-Key. Note that we consider the *stateful* setting which requires that the signer keep a *state* of all previously signed messages and keys.

▶ **Definition 4** ((Stateful) Digital Signatures with Revocable Signing Keys (DSR-Key)). *A (stateful) digital signature scheme with revocable signing keys is the following set of algorithms* (Setup, KeyGen, Sign, Ver, Del, Cert) *consisting of:*

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{ck}, \mathsf{pp})$: *on input the security parameter* λ*, it outputs a classical key* ck *and a classical parameter* pp.
- $\mathsf{KeyGen}(\mathsf{pp}) \to (\mathsf{sigk}_0, \mathsf{vk})$: *on input* pp*, it outputs a quantum signing key* sigk_0 *and a classical verification key* vk.
- $\mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_i, m) \to (\mathsf{sigk}_{i+1}, \sigma)$: *on input* pp*, a message* m *and a signing key* sigk_i*, it outputs a subsequent signing key* sigk_{i+1} *and a classical signature* σ.
- $\mathsf{Ver}(\mathsf{pp}, \mathsf{vk}, m, \sigma) \to \top/\bot$: *on input* pp*,* vk*,* m*, and* σ*, it outputs* $\top/\bot$.
- $\mathsf{Del}(\mathsf{sigk}_i) \to \mathsf{cert}$: *on input* sigk_i*, it outputs a classical certificate* cert.
- $\mathsf{Cert}(\mathsf{pp}, \mathsf{vk}, \mathsf{ck}, \mathsf{cert}, S) \to \top/\bot$: *on input* pp*,* vk*,* ck*,* cert*, and a set* S *consisting of messages, it outputs* $\top/\bot$.

We require the following properties.

Many-time correctness: *For any polynomial* $p = p(\lambda)$*, and any messages* $(m_1, m_2, ..., m_p)$*,*

$$\Pr\left[\bigwedge_{i\in[p]} \top \leftarrow \mathsf{Ver}(\mathsf{pp}, \mathsf{vk}, m_i, \sigma_i) : \begin{array}{r} (\mathsf{pp}, \mathsf{ck}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{sigk}_0, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp}) \\ (\mathsf{sigk}_1, \sigma_1) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_0, m_1) \\ (\mathsf{sigk}_2, \sigma_2) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_1, m_2) \\ \cdots \\ (\mathsf{sigk}_p, \sigma_p) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_{p-1}, m_p) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \quad (4)$$

We say that the scheme satisfies one-time correctness if the above is satisfied for $p = 1$.

EUF-CMA security: *For any QPT adversary* $\mathcal{A}$*,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}(\mathsf{pp}, \mathsf{vk}, m^*, \sigma^*) : \begin{array}{r} (\mathsf{pp}, \mathsf{ck}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{sigk}_0, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp}) \\ (m^*, \sigma^*) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{Sign}}}(\mathsf{vk}) \end{array}\right] \leq \mathsf{negl}(\lambda), \quad (5)$$

where $\mathcal{O}_{\mathsf{Sign}}$ *is a stateful signing oracle defined below and* $\mathcal{A}$ *is not allowed to query the oracle on* m^**:*

$\mathcal{O}_{\mathsf{Sign}}$*: Its initial state is set to be* $(\mathsf{pp}, \mathsf{sigk}_0)$*. When a message* m *is queried, it proceeds as follows:*

- *Parse its state as* $(\mathsf{pp}, \mathsf{sigk}_i)$.
- *Run* $(\mathsf{sigk}_{i+1}, \sigma) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_i, m)$.
- *Return* σ *to* $\mathcal{A}$ *and update its state to* $(\mathsf{pp}, \mathsf{sigk}_{i+1})$.

We say that the scheme satisfies one-time EUF-CMA security if Equation (5) *holds for any* $\mathcal{A}$ *that submits at most one query to the oracle.*

Many-time deletion correctness: *For any polynomial* $p = p(\lambda)$*, and any messages* $(m_1, m_2, ..., m_p)$*, the quantity*

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{pp}, \mathsf{vk}, \mathsf{ck}, \mathsf{cert}, \{m_1, m_2, ..., m_p\}) : \begin{array}{r} (\mathsf{pp}, \mathsf{ck}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{sigk}_0, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp}) \\ (\mathsf{sigk}_1, \sigma_1) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_0, m_1) \\ (\mathsf{sigk}_2, \sigma_2) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_1, m_2) \\ \cdots \\ (\mathsf{sigk}_p, \sigma_p) \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathsf{sigk}_{p-1}, m_p) \\ \mathsf{cert} \leftarrow \mathsf{Del}(\mathsf{sigk}_p) \end{array}\right] \quad (6)$$

is at least $1 - \mathsf{negl}(\lambda)$*. We remark that we require the above to also hold for the case of* $p = 0$*, in which case the fifth component of the input of* Cert *is the empty set* $\emptyset$*. We say that the scheme satisfies one-time deletion correctness if the above property is satisfied for* $p \leq 1$.

Many-time deletion security: *For any QPT adversary $\mathcal{A}$,*

$$\Pr\left[\begin{array}{l} \top \leftarrow \mathsf{Cert}(\mathsf{pp},\mathsf{vk},\mathsf{ck},\mathsf{cert},S) \\ \wedge\, m^* \notin S \\ \wedge\, \top \leftarrow \mathsf{Ver}(\mathsf{pp},\mathsf{vk},m^*,\sigma^*) \end{array} : \begin{array}{c} (\mathsf{pp},\mathsf{ck}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{vk},\mathsf{cert},S,m^*,\sigma^*) \leftarrow \mathcal{A}(\mathsf{pp}) \end{array}\right] \leq \mathsf{negl}(\lambda). \tag{7}$$

We say that the scheme satisfies one-time deletion security if the above property is satisfied if we additionally require $|S| \leq 1$.

▶ Remark 5. Following the definition of single signer security in [4] or copy-protection security in [31], it is also reasonable to define deletion security as follows:

For any pair $(\mathcal{A}_1, \mathcal{A}_2)$ of QPT adversaries and any distribution $\mathcal{D}$ with super-logarithmic min-entropy over the message space, the following probability is negligible in λ:

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{pp},\mathsf{vk},\mathsf{ck},\mathsf{cert},S) \wedge \top \leftarrow \mathsf{Ver}(\mathsf{pp},\mathsf{vk},m,\sigma) : \begin{array}{r} (\mathsf{pp},\mathsf{ck}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\mathsf{vk},\mathsf{cert},S,\mathsf{st}) \leftarrow \mathcal{A}_1(\mathsf{pp}) \\ m \leftarrow \mathcal{D} \\ \sigma \leftarrow \mathcal{A}_2(m,\mathsf{st}) \end{array}\right]. \tag{8}$$

It is easy to see that our definition implies the above, but the converse is unlikely. This is why we define deletion security as in Definition 4.

4.2 One-Time Construction for Single-Bit Messages

In the full version of the paper, we construct one-time DSR-Key for single-bit messages from 2-OSS in a black-box way.

▶ **Theorem 6.** *If two-tier one-shot signatures exist, then digital signatures with revocable signing keys with the message space $\{0,1\}$ that satisfy one-time variants of correctness, EUF-CMA security, deletion correctness, and deletion security in Definition 4 exist.*

4.3 From Single-Bit to Multi-Bit Messages

In the full version of the paper, we also show the following theorem which says that we can expand the message space to $\{0,1\}^*$ using collision-resistant hashes.

▶ **Theorem 7.** *If collision-resistant hash functions and digital signatures with revocable signing keys with the message space $\{0,1\}$ that satisfy one-time variants of correctness, EUF-CMA security, deletion correctness, and deletion security in Definition 4 exist, then a similar scheme with the message space $\{0,1\}^*$ exists.*

The proof of correctness is immediate and the proof of one-time EUF-CMA security follows from standard techniques which allow conventional signature schemes to handle messages of arbitrarily length, see [27] for example. Therefore, it suffices to show that the scheme Σ' satisfies the one-time variants of deletion correctness and deletion security. We show this in the full version.

4.4 From One-Time Schemes to Many-Time Schemes

In this section, we show how to extend any one-time scheme into a proper many-time scheme as in Definition 4. The transformation is inspired by the chain-based approach for constructing many-time digital signatures, see [27] for example.[11]

Let $\mathsf{OT} = (\mathsf{OT.Setup}, \mathsf{OT.KeyGen}, \mathsf{OT.Sign}, \mathsf{OT.Ver}, \mathsf{OT.Del}, \mathsf{OT.Cert})$ be a scheme which satisfies the one-time variants of correctness, EUF-CMA security, deletion correctness, and deletion security according to in Definition 4, and has the message space $\{0,1\}^*$. Then, we construct $\mathsf{MT} = (\mathsf{MT.Setup}, \mathsf{MT.KeyGen}, \mathsf{MT.Sign}, \mathsf{MT.Ver}, \mathsf{MT.Del}, \mathsf{MT.Cert})$ with the message space $\{0,1\}^n$ as follows:

- $\mathsf{MT.Setup}(1^\lambda) \to (\mathsf{ck}, \mathsf{pp})$: This is the same as $\mathsf{OT.Setup}$.
- $\mathsf{MT.KeyGen}(\mathsf{pp}) \to (\mathsf{sigk}, \mathsf{vk})$: run $(\mathsf{ot.sigk}_0, \mathsf{ot.vk}_0) \leftarrow \mathsf{OT.KeyGen}(\mathsf{pp})$ and output $\mathsf{sigk} := \mathsf{ot.sigk}_0$ as the quantum signing key and $\mathsf{vk} := \mathsf{ot.vk}_0$ as the classical verification key.
- $\mathsf{MT.Sign}(\mathsf{pp}, \mathsf{sigk}_i, m) \to (\mathsf{sigk}_{i+1}, \sigma)$: on input the public parameter pp, a quantum signing key sigk_i, and a message $m \in \{0,1\}^n$ proceed as follows:
  1. Parse sigk_i as $(\mathsf{ot.sigk}_i, \{\mathsf{ot.sigk}'_j\}_{j\in\{0,1,\ldots,i-1\}}, \{m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i]})$
  2. Generate $(\mathsf{ot.sigk}_{i+1}, \mathsf{ot.vk}_{i+1}) \leftarrow \mathsf{OT.KeyGen}(\mathsf{pp})$.
  3. Run
  $$(\mathsf{ot.sigk}'_i, \mathsf{ot.}\sigma_{i+1}) \leftarrow \mathsf{OT.Sign}(\mathsf{pp}, \mathsf{ot.sigk}_i, m \| \mathsf{ot.vk}_{i+1}).$$
  4. Set $m_{i+1} := m$ and output a subsequent signing key
  $$\mathsf{sigk}_{i+1} := (\mathsf{ot.sigk}_{i+1}, \{\mathsf{ot.sigk}'_j\}_{j\in\{0,1,\ldots,i\}}, \{m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i+1]})$$
  and a signature
  $$\sigma := \{m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i+1]}.$$
- $\mathsf{MT.Ver}(\mathsf{pp}, \mathsf{vk}, m, \sigma) \to \top/\bot$: on input pp, a key vk, a message m, and signature σ, proceed as follows.
  1. Parse σ as $\{m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i]}$ and let $\mathsf{ot.vk}_0 = \mathsf{vk}$.
  2. Output $\top$ if $m = m_i$ and $\mathsf{OT.Ver}(\mathsf{pp}, \mathsf{ot.vk}_{j-1}, m_j \| \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j) = \top$ for every $j \in [i]$.
- $\mathsf{MT.Del}(\mathsf{sigk}_i) \to \mathsf{cert}$: on input sigk, proceed as follows:
  1. Parse sigk_i as $(\mathsf{ot.sigk}_i, \{\mathsf{ot.sigk}'_j\}_{j\in\{0,1,\ldots,i-1\}}, \{m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i]})$.
  2. For $j \in \{0, 1, \ldots, i-1\}$, run $\mathsf{ot.cert}_j \leftarrow \mathsf{OT.Del}(\mathsf{ot.sigk}'_j)$.
  3. Run $\mathsf{ot.cert}_i \leftarrow \mathsf{OT.Del}(\mathsf{ot.sigk}_i)$.
  4. Output $\mathsf{cert} := \{\mathsf{ot.cert}_j, m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i]}$.
- $\mathsf{MT.Cert}(\mathsf{pp}, \mathsf{vk}, \mathsf{ck}, \mathsf{cert}, S) \to \top/\bot$: on input $\mathsf{pp}, \mathsf{vk}, \mathsf{ck}$, cert, and S, parse the certificate cert as a tuple $\{\mathsf{ot.cert}_j, m_j, \mathsf{ot.vk}_j, \mathsf{ot.}\sigma_j\}_{j\in[i]}$, let $\mathsf{ot.vk}_0 = \mathsf{vk}$, and output $\top$ if the following holds:
  - $S = \{m_1, m_2, \ldots, m_i\}$,
  - $\mathsf{OT.Cert}(\mathsf{ot.vk}_{j-1}, \mathsf{ck}, \mathsf{ot.cert}_{j-1}, \{m_j \| \mathsf{ot.vk}_j\}) = \top$ for every $j \in [i]$, and
  - $\mathsf{OT.Cert}(\mathsf{ot.vk}_i, \mathsf{ck}, \mathsf{ot.cert}_i, \emptyset) = \top$.

In the full version of the paper, we prove the following theorem.

[11] We could also use the tree-based construction [32], which has a shorter (logarithmic) signature length. We describe the chain-based construction here for ease of presentation.

▶ **Theorem 8.** *Suppose that* $(\mathsf{OT.Setup}, \mathsf{OT.KeyGen}, \mathsf{OT.Sign}, \mathsf{OT.Ver}, \mathsf{OT.Del}, \mathsf{OT.Cert})$ *satisfies the one-time variants of correctness, EUF-CMA security, deletion correctness, and deletion security in Definition 4. Then, the "many-time scheme" which consists of the tuple* $(\mathsf{MT.Setup}, \mathsf{MT.KeyGen}, \mathsf{MT.Sign}, \mathsf{MT.Ver}, \mathsf{MT.Del}, \mathsf{MT.Cert})$ *satisfies many-time variants of each of the properties.*

5 Adaptive Hardcore Bit Property for OWFs

In this section, we introduce a new concept, which we call *adaptive hardcore bit property for OWFs*, and show it from the existence of OWFs. This property is inspired by the adaptive hardcore bit property which was shown for a family of noisy trapdoor claw-free functions by Brakerski et al. [13]. Our notion of the adaptive hardcore bit property for OWFs will be used to construct two-tier tokenized signatures.

5.1 Statements

The formal statement of the adaptive hardcore bit property for OWFs is given as follows. (Its proof is given later.)

▶ **Theorem 9** (Adaptive Hardcore Bit Property for OWFs). *Let* $\lambda \in \mathbb{N}$ *be the security parameter and let* $\ell(\lambda), \kappa(\lambda) \in \mathbb{N}$ *be polynomials. Let* $f : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\kappa(\lambda)}$ *be a (quantumly-secure) OWF. Then, for any QPT adversary* $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$*, it holds that*

$$\Pr\left[\begin{array}{c} f(x) \in \{f(x_0), f(x_1)\} \\ \bigwedge \\ d \cdot (x_0 \oplus x_1) = c \end{array} : \begin{array}{l} x_0 \leftarrow \{0,1\}^{\ell(\lambda)},\ x_1 \leftarrow \{0,1\}^{\ell(\lambda)} \\ c \leftarrow \{0,1\} \\ (x,d) \leftarrow \mathcal{A}_\lambda\left(\frac{|x_0\rangle + (-1)^c |x_1\rangle}{\sqrt{2}}, f(x_0), f(x_1)\right) \end{array}\right] \le \frac{1}{2} + \mathsf{negl}(\lambda). \tag{9}$$

We actually use its amplified version, which is given as follows. (Its proof is given later.)

▶ **Theorem 10** (Amplified Adaptive Hardcore Bit Property for OWFs). *Let* $\lambda \in \mathbb{N}$ *be the security parameter and let* $\ell(\lambda), \kappa(\lambda), n(\lambda) \in \mathbb{N}$ *be polynomials. Let* $f : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\kappa(\lambda)}$ *be a (quantumly-secure) OWF. Then, for any QPT adversary* $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$*, it holds that*

$$\Pr\left[\begin{array}{c} \wedge_{i \in [n]} f(x_i) \in \{f(x_i^0), f(x_i^1)\} \\ \bigwedge \\ \wedge_{i \in [n]} d_i \cdot (x_i^0 \oplus x_i^1) = c_i \end{array} : \begin{array}{l} \forall i \in [n] \,:\, x_i^0 \leftarrow \{0,1\}^{\ell(\lambda)},\ x_i^1 \leftarrow \{0,1\}^{\ell(\lambda)} \\ \forall i \in [n] \,:\, c_i \leftarrow \{0,1\} \\ \{x_i, d_i\}_{i \in [n]} \leftarrow \mathcal{A}_\lambda\left(\bigotimes_{i=1}^n \frac{|x_i^0\rangle + (-1)^{c_i} |x_i^1\rangle}{\sqrt{2}}, \{f(x_i^b)\}_{i,b}\right) \end{array}\right] \tag{10}$$

is at most negligible in λ.

5.2 Theorem of [10]

In order to show adaptive hardcore bit property for OWFs, we use the following theorem which is implicit in [10, Theorem 3.1]. The only difference is that we additionally reveal both pre-images as part of the distribution $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda \in \mathbb{N}, b \in \{0,1\}}$. We remark that the proof is the same.

▶ **Theorem 11** (Implicit in [10], Theorem 3.1). *Let* $\lambda \in \mathbb{N}$ *be the security parameter, and let* $\ell(\lambda), \kappa(\lambda) \in \mathbb{N}$ *be polynomials. Let* $f : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\kappa(\lambda)}$ *be a OWF secure against QPT adversaries. Let* $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot,\cdot)\}_{\lambda \in \mathbb{N}}$ *be a quantum operation with four arguments: an* $\ell(\lambda)$*-bit* *string z, two $\kappa(\lambda)$-bit strings y_0, y_1, and an $\ell(\lambda)$-qubit quantum state $|\psi\rangle$. Suppose that for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, $z \in \{0,1\}^{\ell(\lambda)}, y_0, y_1 \in \{0,1\}^{\kappa(\lambda)}$, and $\ell(\lambda)$-qubit state $|\psi\rangle$,*

$$\left|\Pr\left[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(z, y_0, y_1, |\psi\rangle)) = 1\right] - \Pr\left[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(0^{\ell(\lambda)}, y_0, y_1, |\psi\rangle)) = 1\right]\right| = \mathsf{negl}(\lambda).$$

That is, $\mathcal{Z}_\lambda$ is semantically-secure with respect to its first input. Now, for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, consider the distribution $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda\in\mathbb{N}, b\in\{0,1\}}$ over quantum states, obtained by running $\mathcal{A}_\lambda$ as follows.

- *Sample $x_0, x_1 \leftarrow \{0,1\}^{\ell(\lambda)}$, define $y_0 = f(x_0), y_1 = f(x_1)$ and initialize $\mathcal{A}_\lambda$ with*

$$\mathcal{Z}_\lambda\left(x_0 \oplus x_1, y_0, y_1, \frac{|x_0\rangle + (-1)^b |x_1\rangle}{\sqrt{2}}\right).$$

- *$\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{\ell(\lambda)}$ and a residual state on register A'.*
- *If $f(x') \in \{y_0, y_1\}$, then output (x_0, x_1, A'), and otherwise output $\bot$.*

Then, it holds that

$$\left\|\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(0) - \widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(1)\right\|_{\mathrm{tr}} \le \mathsf{negl}(\lambda). \tag{11}$$

We can show the following parallel version. (It can be shown by the standard hybrid argument. A detailed proof is given in the full version of the paper.)

▶ **Theorem 12** (Parallel version of Theorem 11). *Let $\lambda \in \mathbb{N}$ be the security parameter. Let $\ell(\lambda), \kappa(\lambda), n(\lambda) \in \mathbb{N}$ be polynomials. Let $f : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\kappa(\lambda)}$ be a OWF secure against QPT adversaries. Let $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot,\cdot)\}_{\lambda\in\mathbb{N}}$ be a quantum operation with four arguments: an $\ell(\lambda)$-bit string z, two $\kappa(\lambda)$-bit strings y_0, y_1, and an $\ell(\lambda)$-qubit quantum state $|\psi\rangle$. Suppose that for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, $z \in \{0,1\}^{\ell(\lambda)}, y_0, y_1 \in \{0,1\}^{\kappa(\lambda)}$, and $\ell(\lambda)$-qubit state $|\psi\rangle$,*

$$\left|\Pr\left[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(z, y_0, y_1, |\psi\rangle)) = 1\right] - \Pr\left[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(0^{\ell(\lambda)}, y_0, y_1, |\psi\rangle)) = 1\right]\right| = \mathsf{negl}(\lambda).$$

That is, $\mathcal{Z}_\lambda$ is semantically-secure with respect to its first input. Now, for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, consider the distribution $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b_1, ..., b_{n(\lambda)})\right\}_{\lambda\in\mathbb{N}, b_i\in\{0,1\}}$ over quantum states, obtained by running $\mathcal{A}_\lambda$ as follows.

- *Sample $x_i^0, x_i^1 \leftarrow \{0,1\}^{\ell(\lambda)}$ for each $i \in [n(\lambda)]$, define $y_i^0 = f(x_i^0), y_i^1 = f(x_i^1)$ and initialize $\mathcal{A}_\lambda$ with*

$$\bigotimes_{i\in[n(\lambda)]} \mathcal{Z}_\lambda\left(x_i^0 \oplus x_i^1, y_i^0, y_i^1, \frac{|x_i^0\rangle + (-1)^{b_i} |x_i^1\rangle}{\sqrt{2}}\right). \tag{12}$$

- *$\mathcal{A}_\lambda$'s output is parsed as strings $x_i' \in \{0,1\}^{\ell(\lambda)}$ for $i \in [n(\lambda)]$ and a residual state on register A'.*
- *If $f(x_i') \in \{y_i^0, y_i^1\}$ for all $i \in [n(\lambda)]$, output $(\{x_i^0\}_{i\in[n(\lambda)]}, \{x_i^1\}_{i\in[n(\lambda)]}, \mathsf{A}')$, and otherwise output $\bot$.*

Then, there exists a negligible function $\mathsf{negl}(\lambda)$ such that for any $b_1, ..., b_{n(\lambda)} \in \{0,1\}$,

$$\left\|\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b_1, ..., b_{n(\lambda)}) - \widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(0, ..., 0)\right\|_{\mathrm{tr}} \le \mathsf{negl}(\lambda). \tag{13}$$

5.3 Proof of Theorem 9

By using Theorem 11, we can show Theorem 9 as follows. Here, we leverage the fact that any algorithm that simultaneously produces both a valid pre-image of the OWF, as well as a string which leaks information about the relative phase between the respective pre-images, must necessarily violate Theorem 11.

Proof of Theorem 9. Let $\ell(\lambda), \kappa(\lambda) \in \mathbb{N}$ be polynomials, and let $f : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\kappa(\lambda)}$ be a (quantumly-secure) OWF. Suppose there exist a QPT algorithm $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$ and a polynomial $p(\lambda)$ such that, for random $x_0, x_1 \leftarrow \{0,1\}^\ell$ and $c \leftarrow \{0,1\}$, it holds that

$$\Pr\left[\bigwedge_{d\cdot(x_0\oplus x_1)=c}^{f(x)\in\{f(x_0),f(x_1)\}} : (x,d) \leftarrow \mathcal{A}_\lambda\left(\frac{|x_0\rangle + (-1)^c |x_1\rangle}{\sqrt{2}}, f(x_0), f(x_1)\right)\right] \geq \frac{1}{2} + \frac{1}{p(\lambda)} \quad (14)$$

for infinitely many λ. We now show how to construct an algorithm that violates Theorem 11. For simplicity, we define the quantum operation $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot,\cdot)\}_{\lambda\in\mathbb{N}}$ in Theorem 11 as

$$\mathcal{Z}_\lambda\left(x_0 \oplus x_1, f(x_0), f(x_1), \frac{|x_0\rangle + (-1)^c |x_1\rangle}{\sqrt{2}}\right) := \left(f(x_0), f(x_1), \frac{|x_0\rangle + (-1)^c |x_1\rangle}{\sqrt{2}}\right).$$

Evidently, our choice of $\mathcal{Z}_\lambda$ is trivially semantically secure with respect to the first argument. Consider the following QPT algorithm $\mathcal{B}_\lambda$:

1. On input $\left(f(x_0), f(x_1), \frac{|x_0\rangle+(-1)^c|x_1\rangle}{\sqrt{2}}\right)$, run
$$(x, d_c) \leftarrow \mathcal{A}_\lambda\left(\frac{|x_0\rangle + (-1)^c |x_1\rangle}{\sqrt{2}}, f(x_0), f(x_1)\right).$$
2. Output x and assign $|d_c\rangle\langle d_c|$ as the residual state.[12]

Adopting the notation from Theorem 11, we define $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c)$.[13] Consider the following distinguisher that distinguishes $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c)$ for $c \in \{0,1\}$:

1. Get $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c)$ as input.
2. If it is $\bot$, output $\bot$ and abort.
3. Output $d_c \cdot (x_0 \oplus x_1) \pmod 2$.

From Equation (14), there exists a polynomial $p(\lambda)$ such that both $f(x) \in \{f(x_0), f(x_1)\}$ and $d_c \cdot (x_0 \oplus x_1) = c \pmod 2$ occur with probability at least $\frac{1}{2} + \frac{1}{p(\lambda)}$. Thus, the distinguisher can distinguish $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(0)$ and $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(1)$ with probability at least $\frac{1}{2} + \frac{1}{p(\lambda)}$, but this means

$$\left\|\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(0) - \widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(1)\right\|_{\mathrm{tr}} \geq \frac{2}{p(\lambda)}.$$

from Theorem 1. This violates Theorem 11. ◀

5.4 Proof of Theorem 10

In this subsection, we show Theorem 10 by using Theorem 12.

[12] Note that we can think of d_c as a classical mixture (i.e., density matrix) over the randomness of $x_0, x_1 \leftarrow \{0,1\}^\ell$, $c \leftarrow \{0,1\}$ and the internal randomness of the algorithm $\mathcal{A}_\lambda$.

[13] It is, roughly speaking, $|x_0\rangle\langle x_0| \otimes |x_1\rangle\langle x_1| \otimes |d_c\rangle\langle d_c|$ for $c \in \{0,1\}$ when $f(x) \in \{f(x_0), f(x_1)\}$, and is $\bot$ when $f(x) \notin \{f(x_0), f(x_1)\}$.

Proof of Theorem 10. For the sake of contradiction, assume that there is a QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$ such that the following quantity

$$\Pr\left[\begin{array}{c} \wedge_{i\in[n]} f(x_i) \in \{f(x_i^0), f(x_i^1)\} \\ \bigwedge \\ \wedge_{i\in[n]} d_i\cdot(x_i^0\oplus x_i^1) = c_i \end{array} : \begin{array}{l} \forall i\in[n] : x_i^0 \leftarrow \{0,1\}^\ell,\ x_i^1\leftarrow\{0,1\}^\ell, c_i \leftarrow \{0,1\} \\ \{x_i,d_i\}_{i\in[n]} \leftarrow \mathcal{A}_\lambda\left(\bigotimes_{i=1}^n \frac{|x_i^0\rangle+(-1)^{c_i}|x_i^1\rangle}{\sqrt{2}}, \{f(x_i^b)\}_{i,b}\right)\end{array}\right] \tag{15}$$

is at least $1/\mathrm{poly}(\lambda)$ for infinitely many λ. We consider the quantum operation $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot,\cdot)\}_{\lambda\in\mathbb{N}}$ in Theorem 12 as

$$\mathcal{Z}_\lambda\left(x_0\oplus x_1, f(x_0), f(x_1), \frac{|x_0\rangle+(-1)^c|x_1\rangle}{\sqrt{2}}\right) := \left(f(x_0), f(x_1), \frac{|x_0\rangle+(-1)^c|x_1\rangle}{\sqrt{2}}\right), \tag{16}$$

which is trivially semantically secure with respect to the first argument. From such $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\mathcal{Z}_\lambda\}_{\lambda\in\mathbb{N}}$, we construct the following QPT adversary $\{\mathcal{B}_\lambda\}_{\lambda\in\mathbb{N}}$ for fixed each $(c_1,...,c_n)\in\{0,1\}^n$:

1. Get $\{f(x_i^b)\}_{i\in[n],b\in\{0,1\}}$ and $\bigotimes_{i\in[n]}\frac{|x_i^0\rangle+(-1)^{c_i}|x_i^1\rangle}{\sqrt{2}}$ as input.
2. Run $(\{x_i\}_{i\in[n]},\{d_i\}_{i\in[n]}) \leftarrow \mathcal{A}_\lambda\left(\bigotimes_{i=1}^n\frac{|x_i^0\rangle+(-1)^{c_i}|x_i^1\rangle}{\sqrt{2}}, \{f(x_i^b)\}_{i\in[n],b\in\{0,1\}}\right)$.
3. Output $\{x_i\}_{i\in[n]}$. Set its residual state as $\bigotimes_{i\in[n]}|d_i\rangle\langle d_i|$.

Then, by using the notation of Theorem 12, we define $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c_1,...,c_n)$.[14] Let us consider the following QPT distinguisher $\{\mathcal{D}_\lambda\}_{\lambda\in\mathbb{N}}$:

1. Get $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c_1,...,c_n)$ as input.
2. If it is $\bot$, output $\bot$. Otherwise, parse it as $\left(\bigotimes_{i\in[n],b\in\{0,1\}}|x_i^b\rangle\langle x_i^b|\right)\otimes\left(\bigotimes_{i\in[n]}|d_i\rangle\langle d_i|\right)$.
3. Compute $c_i' := d_i\cdot(x_i^0\oplus x_i^1)$ for each $i\in[n]$. Output $\{c_i'\}_{i\in[n]}$.

Then, from Equation (15),

$$\frac{1}{2^n}\sum_{(c_1,...,c_n)\in\{0,1\}^n}\Pr[(c_1,...,c_n)\leftarrow\mathcal{D}(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c_1,...,c_n))] \geq \frac{1}{\mathrm{poly}(\lambda)} \tag{17}$$

for infinitely many λ. Now we show that it contradicts Theorem 12.

If Theorem 12 is correct, there exists a negligible function negl such that

$$\left\|\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c_1,...,c_n) - \widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(0,...,0)\right\|_{\mathrm{tr}} \leq \mathsf{negl}(\lambda) \tag{18}$$

for all $(c_1,...,c_n)\in\{0,1\}^n$. However, in that case, there exists a negligible function negl such that

$$\left|\Pr\left[(c_1,...,c_n)\leftarrow\mathcal{D}(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c_1,...,c_n))\right] - \Pr\left[(c_1,...,c_n)\leftarrow\mathcal{D}(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(0,...,0))\right]\right| \leq \mathsf{negl}(\lambda) \tag{19}$$

for all $(c_1,...,c_n)\in\{0,1\}^n$. Then we have

[14] Roughly speaking, it is $\left(\bigotimes_{i\in[n],b\in\{0,1\}}|x_i^b\rangle\langle x_i^b|\right)\otimes\left(\bigotimes_{i\in[n]}|d_i\rangle\langle d_i|\right)$ if $f(x_i)\in\{f(x_i^0),f(x_i^1)\}$ for all $i\in[n]$, and it is $\bot$ otherwise.

$$\begin{aligned}
\frac{1}{\mathrm{poly}(\lambda)} &\le \frac{1}{2^n}\sum_{(c_1,...,c_n)\in\{0,1\}^n} \Pr[(c_1,...,c_n)\leftarrow \mathcal{D}(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(c_1,...,c_n))] && (20)\\
&\le \frac{1}{2^n}\sum_{(c_1,...,c_n)\in\{0,1\}^n} \Big(\Pr[(c_1,...,c_n)\leftarrow \mathcal{D}(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(0,...,0))] + \mathsf{negl}(\lambda)\Big) && (21)\\
&\le \frac{1}{2^n}\sum_{(c_1,...,c_n)\in\{0,1\}^n} \Pr[(c_1,...,c_n)\leftarrow \mathcal{D}(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{B}_\lambda}(0,...,0))] + \mathsf{negl}(\lambda) && (22)\\
&\le \frac{1}{2^n} + \mathsf{negl}(\lambda) && (23)
\end{aligned}$$

for infinitely many λ, which yields a contradiction. Here, the first inequality is from Equation (17), the second inequality is from Equation (19), and the last inequality is from the fact that $\sum_{(c_1,...,c_n)\in\{0,1\}^n}\Pr[(c_1,...,c_n)\leftarrow\mathcal{A}] = 1$ for any algorithm $\mathcal{A}$. ◀

6 Two-Tier Tokenized Signatures

In this section, we will first give the formal definition of two-tier tokenized signatures (2-TS), and then show that they can be constructed from OWFs. For the construction, we use the (amplified) adaptive hardcore bit property for OWFs (Theorem 10).

6.1 Definition

The formal definition is as follows.

▶ **Definition 13** (Two-Tier Tokenized Signatures (2-TS)). *A two-tier tokenized signature scheme is a tuple* $(\mathsf{KeyGen},\mathsf{StateGen},\mathsf{Sign},\mathsf{Ver}_0,\mathsf{Ver}_1)$ *of algorithms such that*

- $\mathsf{KeyGen}(1^\lambda)\to(\mathsf{sk},\mathsf{pk})$: *It is a QPT algorithm that, on input the security parameter λ, outputs a classical secret key* sk *and a classical public key* pk*.*
- $\mathsf{StateGen}(\mathsf{sk})\to\psi$: *It is a QPT algorithm that, on input* sk*, outputs a quantum state ψ.*
- $\mathsf{Sign}(\psi,m)\to\sigma$: *It is a QPT algorithm that, on input ψ and a message $m\in\{0,1\}$, outputs a classical signature σ.*
- $\mathsf{Ver}_0(\mathsf{pk},\sigma)\to\top/\bot$: *It is a QPT algorithm that, on input* pk *and σ, outputs $\top/\bot$.*
- $\mathsf{Ver}_1(\mathsf{sk},\sigma)\to\top/\bot$: *It is a QPT algorithm that, on input* sk *and σ, outputs $\top/\bot$.*

We require the following properties.

Correctness:

$$\Pr\left[\top\leftarrow\mathsf{Ver}_0(\mathsf{pk},\sigma) : \begin{array}{r}(\mathsf{sk},\mathsf{pk})\leftarrow\mathsf{KeyGen}(1^\lambda)\\ \psi\leftarrow\mathsf{StateGen}(\mathsf{sk})\\ \sigma\leftarrow\mathsf{Sign}(\psi,0)\end{array}\right]\ge 1-\mathsf{negl}(\lambda) \tag{24}$$

and

$$\Pr\left[\top\leftarrow\mathsf{Ver}_1(\mathsf{sk},\sigma) : \begin{array}{r}(\mathsf{sk},\mathsf{pk})\leftarrow\mathsf{KeyGen}(1^\lambda)\\ \psi\leftarrow\mathsf{StateGen}(\mathsf{sk})\\ \sigma\leftarrow\mathsf{Sign}(\psi,1)\end{array}\right]\ge 1-\mathsf{negl}(\lambda). \tag{25}$$

Security: *For any QPT adversary $\mathcal{A}$,*

$$\Pr\left[\top\leftarrow\mathsf{Ver}_0(\mathsf{pk},\sigma_0)\wedge\top\leftarrow\mathsf{Ver}_1(\mathsf{sk},\sigma_1) : \begin{array}{r}(\mathsf{sk},\mathsf{pk})\leftarrow\mathsf{KeyGen}(1^\lambda)\\ \psi\leftarrow\mathsf{StateGen}(\mathsf{sk})\\ (\sigma_0,\sigma_1)\leftarrow\mathcal{A}(\psi,\mathsf{pk})\end{array}\right]\le\mathsf{negl}(\lambda). \tag{26}$$

We can show that the following type of security, which we call *one-wayness*, is also satisfied by two-tier tokenized signatures.

▶ **Lemma 14** (One-wayness of two-tier tokenized signatures). *For any QPT adversary $\mathcal{A}$,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}_0(\mathsf{pk},\sigma) : \begin{array}{r}(\mathsf{sk},\mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)\\ \psi \leftarrow \mathcal{A}(\mathsf{pk})\\ \sigma \leftarrow \mathsf{Sign}(\psi,0)\end{array}\right] \le \mathsf{negl}(\lambda). \tag{27}$$

Proof. Assume that there exists a QPT adversary $\mathcal{A}$ such that

$$\Pr\left[\top \leftarrow \mathsf{Ver}_0(\mathsf{pk},\sigma) : \begin{array}{r}(\mathsf{sk},\mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)\\ \psi \leftarrow \mathcal{A}(\mathsf{pk})\\ \sigma \leftarrow \mathsf{Sign}(\psi,0)\end{array}\right] \ge \frac{1}{\mathrm{poly}(\lambda)} \tag{28}$$

for infinitely many λ. Then, from such $\mathcal{A}$, we can construct a QPT adversary $\mathcal{B}$ that breaks the security of the two-tier tokenized signature scheme as follows:

1. Get ψ and pk as input.
2. Run $\psi' \leftarrow \mathcal{A}(\mathsf{pk})$.
3. Run $\sigma_0 \leftarrow \mathsf{Sign}(\psi',0)$ and $\sigma_1 \leftarrow \mathsf{Sign}(\psi,1)$.
4. Output (σ_0,σ_1).

It is clear that $\mathcal{B}$ breaks the security of the two-tier tokenized signature scheme. ◀

6.2 Construction

We show that 2-TS can be constructed from OWFs.

▶ **Theorem 15.** *If OWFs exist, then two-tier tokenized signatures exist.*

Proof. Let f be a OWF. From it, we construct a two-tier tokenized signature scheme as follows:

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sk},\mathsf{pk})$: Choose $x_i^0, x_i^1 \leftarrow \{0,1\}^\ell$ for each $i \in [n]$. Choose $c_i \leftarrow \{0,1\}$ for each $i \in [n]$. Output $\mathsf{sk} := \{c_i, x_i^0, x_i^1\}_{i\in[n]}$ and $\mathsf{pk} := \{f(x_i^0), f(x_i^1)\}_{i\in[n]}$.
- $\mathsf{StateGen}(\mathsf{sk}) \to \psi$: Parse $\mathsf{sk} = \{c_i, x_i^0, x_i^1\}_{i\in[n]}$. Output $\psi := \bigotimes_{i\in[n]} \frac{|x_i^0\rangle + (-1)^{c_i}|x_i^1\rangle}{\sqrt{2}}$.
- $\mathsf{Sign}(\psi,m) \to \sigma$: If $m=0$, measure ψ in the computational basis to get the result $\{z_i\}_{i\in[n]}$ (where $z_i \in \{0,1\}^\ell$ for each $i \in [n]$), and output it as σ. If $m=1$, measure ψ in the Hadamard basis to get the result $\{d_i\}_{i\in[n]}$ (where $d_i \in \{0,1\}^\ell$ for each $i \in [n]$), and output it as σ.
- $\mathsf{Ver}_0(\mathsf{pk},\sigma) \to \top/\bot$: Parse $\mathsf{pk} = \{f(x_i^0), f(x_i^1)\}_{i\in[n]}$ and $\sigma = \{z_i\}_{i\in[n]}$. If $f(z_i) \in \{f(x_i^0), f(x_i^1)\}$ for all $i \in [n]$, output $\top$. Otherwise, output $\bot$.
- $\mathsf{Ver}_1(\mathsf{sk},\sigma) \to \top/\bot$: Parse $\mathsf{sk} = \{c_i, x_i^0, x_i^1\}_{i\in[n]}$ and $\sigma = \{d_i\}_{i\in[n]}$. If $d_i \cdot (x_i^0 \oplus x_i^1) = c_i$ for all $i \in [n]$, output $\top$. Otherwise, output $\bot$.

The correctness is clear. The security is also clear from Theorem 10. ◀

7 Digital Signatures with Revocable Signatures

In this section, we define digital signatures with revocable signatures (DSR-Sign). We also show that it can be constructed from 2-TS, and therefore from OWFs.

7.1 Definition

We first give its formal definition as follows.

▶ **Definition 16** (Digital Signatures with Revocable Signatures (DSR-Sign)). *A digital signature scheme with revocable signatures is a set* $(\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Ver}, \mathsf{Del}, \mathsf{Cert})$ *of algorithms that satisfy the following.*

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sigk}, \mathsf{vk})$: *It is a QPT algorithm that, on input the security parameter* λ*, outputs a classical signing key* sigk *and a classical public verification key* vk*.*
- $\mathsf{Sign}(\mathsf{sigk}, m) \to (\psi, \mathsf{ck})$: *It is a QPT algorithm that, on input a message* m *and* sigk*, outputs a quantum signature* ψ *and a classical check key* ck*.*
- $\mathsf{Ver}(\mathsf{vk}, \psi, m) \to \top/\bot$: *It is a QPT algorithm that, on input* vk*,* m*, and* ψ*, outputs* $\top/\bot$*.*
- $\mathsf{Del}(\psi) \to \mathsf{cert}$: *It is a QPT algorithm that, on input* ψ*, outputs a classical certificate* cert.
- $\mathsf{Cert}(\mathsf{ck}, \mathsf{cert}) \to \top/\bot$: *It is a QPT algorithm that, on input* ck *and* cert*, outputs* $\top/\bot$*.*

We require the following properties.

Correctness: *For any message* m*,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}(\mathsf{vk}, \psi, m) : \begin{array}{r} (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ (\psi, \mathsf{ck}) \leftarrow \mathsf{Sign}(\mathsf{sigk}, m) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \tag{29}$$

Deletion correctness: *For any message* m*,*

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{ck}, \mathsf{cert}) : \begin{array}{r} (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ (\psi, \mathsf{ck}) \leftarrow \mathsf{Sign}(\mathsf{sigk}, m) \\ \mathsf{cert} \leftarrow \mathsf{Del}(\psi) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \tag{30}$$

Many-time deletion security: *For any adversary* $\mathcal{A}$ *consisting of a pair of QPT algorithms* $(\mathcal{A}_1, \mathcal{A}_2)$*:*

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{ck}^*, \mathsf{cert}) \wedge \top \leftarrow \mathsf{Ver}(\mathsf{vk}, \psi, m^*) : \begin{array}{r} (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ (m^*, \mathsf{st}) \leftarrow \mathcal{A}_1^{\mathsf{Sign}(\mathsf{sigk},\cdot)}(\mathsf{vk}) \\ (\psi^*, \mathsf{ck}^*) \leftarrow \mathsf{Sign}(\mathsf{sigk}, m^*) \\ (\mathsf{cert}, \psi) \leftarrow \mathcal{A}_2^{\mathsf{Sign}(\mathsf{sigk},\cdot)}(\mathsf{st}, \psi^*) \end{array}\right] \leq \mathsf{negl}(\lambda), \tag{31}$$

where $\mathcal{A}$ *is not allowed to query* m^* *to the signing oracle.*

▶ Remark 17. The above definition does not capture the situation where the adversary gets more than one signatures on m^* but deleted all of them. Actually, our construction seems to also satisfy security in such a setting. However, we choose to not formalize it for simplicity.

▶ Remark 18. We can define the standard EUF-CMA security as follows, but it is trivially implied by many-time deletion security, and therefore we do not include EUF-CMA security in the definition of digital signatures with revocable signatures.

▶ Definition 19 (EUF-CMA Security). *For any QPT adversary* $\mathcal{A}$*,*

$$\Pr\left[\top \leftarrow \mathsf{Ver}(\mathsf{vk}, \psi^*, m^*) : \begin{array}{r} (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ (m^*, \psi^*) \leftarrow \mathcal{A}^{\mathsf{Sign}(\mathsf{sigk},\cdot)}(\mathsf{vk}) \end{array}\right] \leq \mathsf{negl}(\lambda), \tag{32}$$

where $\mathcal{A}$ *is not allowed to query* m^* *to the signing oracle.*

We define a weaker version of many-time deletion security, which we call no-query deletion security as follows.

▶ **Definition 20** (No-Query Deletion Security). *It is the same as many-time deletion security, Equation* (31)*, except that $\mathcal{A}$ cannot query the signing oracle.*

The no-query security notion actually implies the many-time case:

▶ **Lemma 21** (Many-Time Deletion Security from No-Query Deletion Security). *Assume that EUF-CMA secure digital signature schemes exist. Then following holds: if there exists a digital signature scheme with revocable signatures which satisfies no-query deletion security, then there is a scheme that satisfies many-time deletion security.*

Proof. Let $(\mathsf{NQ.KeyGen}, \mathsf{NQ.Sign}, \mathsf{NQ.Ver}, \mathsf{NQ.Del}, \mathsf{NQ.Cert})$ be a digital signature scheme with revocable signatures that satisfies no-query deletion security. Let $(\mathsf{MT.KeyGen}, \mathsf{MT.Sign}, \mathsf{MT.Ver})$ be a plain EUF-CMA secure digital signature scheme. From them, we can construct a digital signature scheme $\Sigma := (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Ver}, \mathsf{Del}, \mathsf{Cert})$ with revocable signatures that satisfies many-time deletion security as follows.

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sigk}, \mathsf{vk})$: Run $(\mathsf{mt.sigk}, \mathsf{mt.vk}) \leftarrow \mathsf{MT.KeyGen}(1^\lambda)$. Output $\mathsf{sigk} := \mathsf{mt.sigk}$ and $\mathsf{vk} := \mathsf{mt.vk}$.
- $\mathsf{Sign}(\mathsf{sigk}, m) \to (\psi, \mathsf{ck})$: Parse $\mathsf{sigk} = \mathsf{mt.sigk}$. Run $(\mathsf{nq.sigk}, \mathsf{nq.vk}) \leftarrow \mathsf{NQ.KeyGen}(1^\lambda)$. Run $(\phi, \mathsf{nq.ck}) \leftarrow \mathsf{NQ.Sign}(\mathsf{nq.sigk}, m)$. Run $\sigma \leftarrow \mathsf{MT.Sign}(\mathsf{mt.sigk}, \mathsf{nq.vk} \| m)$. Output $\psi := (\phi, \sigma, \mathsf{nq.vk})$ and $\mathsf{ck} := \mathsf{nq.ck}$.
- $\mathsf{Ver}(\mathsf{vk}, \psi, m) \to \top/\bot$: Parse $\mathsf{vk} = \mathsf{mt.vk}$ and $\psi = (\phi, \sigma, \mathsf{nq.vk})$. Run $\mathsf{MT.Ver}(\mathsf{mt.vk}, \sigma, \mathsf{nq.vk} \| m)$. If the output is $\bot$, output $\bot$ and abort. Run $\mathsf{NQ.Ver}(\mathsf{nq.vk}, \phi, m)$. If the output is $\top$, output $\top$. Otherwise, output $\bot$.
- $\mathsf{Del}(\psi) \to \mathsf{cert}$: Parse $\psi = (\phi, \sigma, \mathsf{nq.vk})$. Run $\mathsf{cert}' \leftarrow \mathsf{NQ.Del}(\phi)$. Output $\mathsf{cert} := \mathsf{cert}'$.
- $\mathsf{Cert}(\mathsf{ck}, \mathsf{cert}) \to \top/\bot$: Parse $\mathsf{ck} = \mathsf{nq.ck}$. Run $\mathsf{NQ.Cert}(\mathsf{nq.ck}, \mathsf{cert})$, and output its output.

We show that Σ satisfies many-time deletion security. In other words, we show that if the many-time deletion security of Σ is broken, then either the no-query deletion security of the digital signature scheme NQ is broken or the EUF-CMA security of the digital signature scheme MT is broken. Assume that there exists a pair of QPT algorithms $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ such that

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{ck}^*, \mathsf{cert}) \wedge \top \leftarrow \mathsf{Ver}(\mathsf{vk}, \psi, m^*) : \begin{array}{r} (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ (m^*, \mathsf{st}) \leftarrow \mathcal{A}_1^{\mathsf{Sign}(\mathsf{sigk}, \cdot)}(\mathsf{vk}) \\ (\psi^*, \mathsf{ck}^*) \leftarrow \mathsf{Sign}(\mathsf{sigk}, m^*) \\ (\mathsf{cert}, \psi) \leftarrow \mathcal{A}_2^{\mathsf{Sign}(\mathsf{sigk}, \cdot)}(\mathsf{st}, \psi^*) \end{array}\right] \geq \frac{1}{\mathrm{poly}(\lambda)} \tag{33}$$

for infinitely many λ, where $\mathcal{A}$ is not allowed to query m^* to the signing oracle. From such $\mathcal{A}$, we construct a QPT adversary $\mathcal{B}$ that breaks the no-query deletion security of the scheme NQ as follows: Let $\mathcal{C}$ be the challenger of the security game of the no-query deletion security.

1. $\mathcal{C}$ runs $(\mathsf{nq.sigk}^*, \mathsf{nq.vk}^*) \leftarrow \mathsf{NQ.KeyGen}(1^\lambda)$.
2. $\mathcal{C}$ sends $\mathsf{nq.vk}^*$ to $\mathcal{B}$.
3. $\mathcal{B}$ runs $(\mathsf{mt.sigk}, \mathsf{mt.vk}) \leftarrow \mathsf{MT.KeyGen}(1^\lambda)$.
4. $\mathcal{B}$ runs $(m^*, \mathsf{st}) \leftarrow \mathcal{A}_1^{\mathsf{Sign}(\mathsf{sigk}, \cdot)}(\mathsf{mt.vk})$. When $\mathcal{A}_1$ queries m to the signing oracle, $\mathcal{B}$ simulates it as follows:
   a. Run $(\mathsf{nq.sigk}, \mathsf{nq.vk}) \leftarrow \mathsf{NQ.KeyGen}(1^\lambda)$.
   b. Run $(\phi, \mathsf{nq.ck}) \leftarrow \mathsf{NQ.Sign}(\mathsf{nq.sigk}, m)$.
   c. Run $\sigma \leftarrow \mathsf{MT.Sign}(\mathsf{mt.sigk}, \mathsf{nq.vk} \| m)$.
   d. Output $\psi := (\phi, \sigma, \mathsf{nq.vk})$ and $\mathsf{ck} := \mathsf{nq.ck}$.
5. $\mathcal{B}$ sends m^* to $\mathcal{C}$.
6. $\mathcal{C}$ runs $(\phi^*, \mathsf{nq.ck}^*) \leftarrow \mathsf{NQ.Sign}(\mathsf{nq.sigk}^*, m^*)$, and sends ϕ^* to $\mathcal{B}$.

7. $\mathcal{B}$ runs $\sigma^* \leftarrow \mathsf{MT.Sign}(\mathsf{mt.sigk}, \mathsf{nq.vk}^*\|m^*)$.
8. $\mathcal{B}$ runs $(\mathsf{cert}, \psi) \leftarrow \mathcal{A}_2^{\mathsf{Sign}(\mathsf{sigk},\cdot)}((\phi^*, \sigma^*, \mathsf{nq.vk}^*))$. When $\mathcal{A}_2$ queries m to the signing oracle, $\mathcal{B}$ simulates it as follows:
   a. Run $(\mathsf{nq.sigk}, \mathsf{nq.vk}) \leftarrow \mathsf{NQ.KeyGen}(1^\lambda)$.
   b. Run $(\phi, \mathsf{nq.ck}) \leftarrow \mathsf{NQ.Sign}(\mathsf{nq.sigk}, m)$.
   c. Run $\sigma \leftarrow \mathsf{MT.Sign}(\mathsf{mt.sigk}, \mathsf{nq.vk}\|m)$.
   d. Output $\psi := (\phi, \sigma, \mathsf{nq.vk})$ and $\mathsf{ck} := \mathsf{nq.ck}$.
9. Parse $\psi = (\phi, \sigma, \eta)$. $\mathcal{B}$ outputs cert and ϕ.

Due to the EUF-CMA security of the scheme MT, $\top \leftarrow \mathsf{MT.Ver}(\mathsf{mt.vk}, \sigma, \eta\|m^*)$ occurs only when $\eta = \mathsf{nq.vk}^*$ except for a negligible probability. Therefore, Equation (33) means that both $\Pr[\top \leftarrow \mathsf{NQ.Ver}(\mathsf{nq.vk}^*, \phi, m^*)]$ and $\Pr[\top \leftarrow \mathsf{NQ.Cert}(\mathsf{nq.ck}^*, \mathsf{cert})]$ are non-negligible for the above $\mathcal{B}$, which breaks the no-query deletion security of the scheme NQ. ◀

7.2 Construction

Here we show the following result.

▶ **Theorem 22.** *If two-tier tokenized signatures exist, then digital signatures with revocable signatures that satisfy no-query deletion security exist.*

From Lemma 21, it also means the following:

▶ **Corollary 23.** *Digital signatures with revocable signatures (that satisfy many-time deletion security) exist if two-tier tokenized signatures and EUF-CMA secure digital signatures exist.*

Proof of Theorem 22. Here, we construct the scheme for the single-bit message space. It is clear that this can be extended to any fixed multi-bit message space case by the parallel execution of the protocol. Moreover, by using universal one-way hash functions, it can be extended to unbounded poly-length message space case [34].

Let $(\mathsf{TS.KeyGen}, \mathsf{TS.StateGen}, \mathsf{TS.Sign}, \mathsf{TS.Ver}_0, \mathsf{TS.Ver}_1)$ be a two-tier tokenized signature scheme. From it, we construct a digital signature scheme with revocable signatures $\Sigma := (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Ver}, \mathsf{Del}, \mathsf{Cert})$ that satisfies no-query deletion security for the single bit message space as follows.

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sigk}, \mathsf{vk})$: Run $(\mathsf{sk}_0, \mathsf{pk}_0) \leftarrow \mathsf{TS.KeyGen}(1^\lambda)$. Run $(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{TS.KeyGen}(1^\lambda)$. Output $\mathsf{sigk} := (\mathsf{sk}_0, \mathsf{sk}_1)$ and $\mathsf{vk} := (\mathsf{pk}_0, \mathsf{pk}_1)$.
- $\mathsf{Sign}(\mathsf{sigk}, m) \to (\psi, \mathsf{ck})$: Parse $\mathsf{sigk} = (\mathsf{sk}_0, \mathsf{sk}_1)$. Run $\psi' \leftarrow \mathsf{TS.StateGen}(\mathsf{sk}_m)$. Output $\psi := \psi'$ and $\mathsf{ck} := \mathsf{sk}_m$.
- $\mathsf{Ver}(\mathsf{vk}, \psi, m) \to \top/\bot$: Parse $\mathsf{vk} := (\mathsf{pk}_0, \mathsf{pk}_1)$. Run $\sigma \leftarrow \mathsf{TS.Sign}(\psi, 0)$. Run $\mathsf{TS.Ver}_0(\mathsf{pk}_m, \sigma)$, and output its output.[15]
- $\mathsf{Del}(\psi) \to \mathsf{cert}$: Run $\sigma \leftarrow \mathsf{TS.Sign}(\psi, 1)$, and output $\mathsf{cert} := \sigma$.
- $\mathsf{Cert}(\mathsf{ck}, \mathsf{cert}) \to \top/\bot$: Parse $\mathsf{ck} = \mathsf{sk}_m$. Run $\mathsf{TS.Ver}_1(\mathsf{sk}_m, \mathsf{cert})$, and output its output.

Correctness and the deletion correctness are clear. Let us show the no-query deletion security. Assume that there is a pair of QPT algorithms $(\mathcal{A}_1, \mathcal{A}_2)$ such that

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{ck}^*, \mathsf{cert}) \wedge \top \leftarrow \mathsf{Ver}(\mathsf{vk}, \psi, m^*) : \begin{array}{r} (\mathsf{sigk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ (m^*, \mathsf{st}) \leftarrow \mathcal{A}_1(\mathsf{vk}) \\ (\psi^*, \mathsf{ck}^*) \leftarrow \mathsf{Sign}(\mathsf{sigk}, m^*) \\ (\mathsf{cert}, \psi) \leftarrow \mathcal{A}_2(\mathsf{st}, \psi^*) \end{array}\right] \geq \frac{1}{\mathrm{poly}(\lambda)} \qquad (34)$$

[15] The verification algorithm destroys the signature, but it can be done in a non-destructive way by coherently applying this procedure and then doing the uncomputation.

for infinitely many λ. From $\mathcal{A}$, we can construct a QPT adversary $\mathcal{B}$ that breaks the original two-tier tokenized signature scheme as follows:

1. Get ψ^* and pk as input.
2. Run $(\mathsf{sk}', \mathsf{pk}') \leftarrow \mathsf{TS.KeyGen}(1^\lambda)$. Choose $r \leftarrow \{0,1\}$. If $r = 0$, set $\mathsf{vk} := (\mathsf{pk}, \mathsf{pk}')$. If $r = 1$, set $\mathsf{vk} := (\mathsf{pk}', \mathsf{pk})$.
3. Run $(m^*, \mathsf{st}) \leftarrow \mathcal{A}_1(\mathsf{vk})$. If $r \neq m^*$, output $\perp$ and abort.
4. Run $(\mathsf{cert}, \psi) \leftarrow \mathcal{A}_2(\mathsf{st}, \psi^*)$.
5. Run $\sigma_0 \leftarrow \mathsf{TS.Sign}(\psi, 0)$. Define $\sigma_1 := \mathsf{cert}$.
6. Output (σ_0, σ_1).

It is clear that $\Pr[\mathcal{B}$ breaks the two-tier tokenized signature scheme$] \geq \frac{1}{2}\Pr[\mathcal{A}$ breaks $\Sigma]$. Therefore, from Equation (34), $\mathcal{B}$ breaks the two-tier tokenized signature scheme. ◀

References

1 Scott Aaronson. Quantum copy-protection and quantum money. In *Proceedings of the 24th Annual IEEE Conference on Computational Complexity, CCC 2009, Paris, France, 15-18 July 2009*, pages 229–242. IEEE Computer Society, 2009. doi:10.1109/CCC.2009.42.

2 Scott Aaronson, Jiahui Liu, Qipeng Liu, Mark Zhandry, and Ruizhe Zhang. New approaches for quantum copy-protection. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology – CRYPTO 2021, Part I*, volume 12825 of *Lecture Notes in Computer Science*, pages 526–555, Virtual Event, August 16–20 2021. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-84242-0_19.

3 Shweta Agrawal, Fuyuki Kitagawa, Ryo Nishimaki, Shota Yamada, and Takashi Yamakawa. Public key encryption with secure key leasing. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023, Part I*, volume 14004 of *Lecture Notes in Computer Science*, pages 581–610, Lyon, France, April 23–27 2023. Springer, Heidelberg, Germany. doi:10.1007/978-3-031-30545-0_20.

4 Ryan Amos, Marios Georgiou, Aggelos Kiayias, and Mark Zhandry. One-shot signatures and applications to hybrid quantum/classical authentication. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *52nd Annual ACM Symposium on Theory of Computing*, pages 255–268, Chicago, IL, USA, June 22–26 2020. ACM Press. doi:10.1145/3357713.3384304.

5 Prabhanjan Ananth and Rolando L. La Placa. Secure software leasing. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021, Part II*, volume 12697 of *Lecture Notes in Computer Science*, pages 501–530, Zagreb, Croatia, October 17–21 2021. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-77886-6_17.

6 Prabhanjan Ananth, Alexander Poremba, and Vinod Vaikuntanathan. Revocable cryptography from learning with errors. In Guy Rothblum and Hoeteck Wee, editors, *Theory of Cryptography*, pages 93–122, Cham, 2023. Springer Nature Switzerland.

7 James Bartusek, Sanjam Garg, Vipul Goyal, Dakshita Khurana, Giulio Malavolta, Justin Raizes, and Bhaskar Roberts. Obfuscation and outsourced computation with certified deletion. Cryptology ePrint Archive, Paper 2023/265, 2023. URL: https://eprint.iacr.org/2023/265.

8 James Bartusek and Dakshita Khurana. Cryptography with certified deletion. Cryptology ePrint Archive, Paper 2022/1178, 2022. URL: https://eprint.iacr.org/2022/1178.

9 James Bartusek, Dakshita Khurana, Giulio Malavolta, Alexander Poremba, and Michael Walter. Weakening assumptions for publicly-verifiable deletion. Cryptology ePrint Archive, Paper 2023/559, 2023. URL: https://eprint.iacr.org/2023/559.

10 James Bartusek, Dakshita Khurana, Giulio Malavolta, Alexander Poremba, and Michael Walter. Weakening assumptions for publicly-verifiable deletion. In *Theory of Cryptography: 21st International Conference, TCC 2023, Taipei, Taiwan, November 29–December 2, 2023, Proceedings, Part IV*, pages 183–197, Berlin, Heidelberg, 2023. Springer-Verlag. doi:10.1007/978-3-031-48624-1_7.

11 James Bartusek, Dakshita Khurana, and Alexander Poremba. Publicly-verifiable deletion via target-collapsing functions. In *Advances in Cryptology – CRYPTO 2023: 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20–24, 2023, Proceedings, Part V*, pages 99–128, Berlin, Heidelberg, 2023. Springer-Verlag. doi:10.1007/978-3-031-38554-4_4.

12 Shalev Ben-David and Or Sattath. Quantum tokens for digital signatures. *Quantum*, 2023.

13 Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. *Journal of the ACM*, 68(5):31:1–31:47, 2021.

14 Anne Broadbent and Rabib Islam. Quantum encryption with certified deletion. In Rafael Pass and Krzysztof Pietrzak, editors, *TCC 2020: 18th Theory of Cryptography Conference, Part III*, volume 12552 of *Lecture Notes in Computer Science*, pages 92–122, Durham, NC, USA, November 16–19 2020. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-64381-2_4.

15 Anne Broadbent and Sébastien Lord. Uncloneable quantum encryption via oracles. In Steven T. Flammia, editor, *15th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2020, June 9-12, 2020, Riga, Latvia*, volume 158 of *LIPIcs*, pages 4:1–4:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.TQC.2020.4.

16 Orestis Chardouvelis, Vipul Goyal, Aayush Jain, and Jiahui Liu. Quantum key leasing for pke and fhe with a classical lessor. Cryptology ePrint Archive, Paper 2023/1640, 2023. URL: https://eprint.iacr.org/2023/1640.

17 Andrea Coladangelo, Jiahui Liu, Qipeng Liu, and Mark Zhandry. Hidden cosets and applications to unclonable cryptography. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology – CRYPTO 2021, Part I*, volume 12825 of *Lecture Notes in Computer Science*, pages 556–584, Virtual Event, August 16–20 2021. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-84242-0_20.

18 Andrea Coladangelo, Christian Majenz, and Alexander Poremba. Quantum copy-protection of compute-and-compare programs in the quantum random oracle model. Cryptology ePrint Archive, Paper 2020/1194, 2020. URL: https://eprint.iacr.org/2020/1194.

19 Marios Georgiou and Mark Zhandry. Unclonable decryption keys. Cryptology ePrint Archive, Paper 2020/877, 2020. URL: https://eprint.iacr.org/2020/877.

20 Daniel Gottesman. Unclonable encryption, 2002. arXiv:0210062.

21 Carl W. Helstrom. Quantum detection and estimation theory. *Journal of Statistical Physics*, 1:231–252, 1969.

22 Taiga Hiroka, Fuyuki Kitagawa, Tomoyuki Morimae, Ryo Nishimaki, Tapas Pal, and Takashi Yamakawa. Certified everlasting secure collusion-resistant functional encryption, and more. Cryptology ePrint Archive, Paper 2023/236, 2023. URL: https://eprint.iacr.org/2023/236.

23 Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum encryption with certified deletion, revisited: Public key, attribute-based, and classical communication. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2021, Part I*, volume 13090 of *Lecture Notes in Computer Science*, pages 606–636, Singapore, December 6–10 2021. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-92062-3_21.

24 Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Certified everlasting zero-knowledge proof for QMA. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part I*, volume 13507 of *Lecture Notes in Computer Science*, pages 239–268, Santa Barbara, CA, USA, August 15–18 2022. Springer, Heidelberg, Germany. doi:10.1007/978-3-031-15802-5_9.

25 A.S Holevo. Statistical decision theory for quantum systems. *Journal of Multivariate Analysis*, 3(4):337–394, 1973. doi:10.1016/0047-259X(73)90028-6.

26 Zhengfeng Ji, Youming Qiao, Fang Song, and Aaram Yun. General linear group action on tensors: A candidate for post-quantum cryptography. In Dennis Hofheinz and Alon Rosen, editors, *TCC 2019: 17th Theory of Cryptography Conference, Part I*, volume 11891 of *Lecture Notes in Computer Science*, pages 251–281, Nuremberg, Germany, December 1–5 2019. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-36030-6_11.

27 Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. Chapman and Hall/CRC Press, 2007. URL: http://www.cs.umd.edu/~jkatz/imc.html.

28 Fuyuki Kitagawa and Ryo Nishimaki. Functional encryption with secure key leasing. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022, Part IV*, volume 13794 of *Lecture Notes in Computer Science*, pages 569–598, Taipei, Taiwan, December 5–9 2022. Springer, Heidelberg, Germany. doi:10.1007/978-3-031-22972-5_20.

29 Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Secure software leasing from standard assumptions. In Kobbi Nissim and Brent Waters, editors, *TCC 2021: 19th Theory of Cryptography Conference, Part I*, volume 13042 of *Lecture Notes in Computer Science*, pages 31–61, Raleigh, NC, USA, November 8–11 2021. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-90459-3_2.

30 Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Publicly verifiable deletion from minimal assumptions. Cryptology ePrint Archive, Paper 2023/538, 2023. URL: https://eprint.iacr.org/2023/538.

31 Jiahui Liu, Qipeng Liu, Luowen Qian, and Mark Zhandry. Collusion resistant copy-protection for watermarkable functionalities. In Eike Kiltz and Vinod Vaikuntanathan, editors, *TCC 2022: 20th Theory of Cryptography Conference, Part I*, volume 13747 of *Lecture Notes in Computer Science*, pages 294–323, Chicago, IL, USA, November 7–10 2022. Springer, Heidelberg, Germany. doi:10.1007/978-3-031-22318-1_11.

32 Ralph C. Merkle. A digital signature based on a conventional encryption function. In Carl Pomerance, editor, *Advances in Cryptology – CRYPTO'87*, volume 293 of *Lecture Notes in Computer Science*, pages 369–378, Santa Barbara, CA, USA, August 16–20 1988. Springer, Heidelberg, Germany. doi:10.1007/3-540-48184-2_32.

33 Tomoyuki Morimae and Takashi Yamakawa. Classically verifiable NIZK for QMA with preprocessing. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022, Part IV*, volume 13794 of *Lecture Notes in Computer Science*, pages 599–627, Taipei, Taiwan, December 5–9 2022. Springer, Heidelberg, Germany. doi:10.1007/978-3-031-22972-5_21.

34 Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *21st Annual ACM Symposium on Theory of Computing*, pages 33–43, Seattle, WA, USA, May 15–17 1989. ACM Press. doi:10.1145/73007.73011.

35 Alexander Poremba. Quantum proofs of deletion for learning with errors. Cryptology ePrint Archive, Paper 2022/295, 2022. URL: https://eprint.iacr.org/2022/295.

36 Alexander Poremba. Quantum Proofs of Deletion for Learning with Errors. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 90:1–90:14, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ITCS.2023.90.

37 Roy Radian and Or Sattath. Semi-quantum money. *arXiv/1908.08889*, 2019. arXiv:1908.08889.

38 Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, pages 84–93. ACM Press, 2005.

39 Ronald L. Rivest. Can we eliminate certificate revocation lists? In Rafael Hirschfeld, editor, *Proceedings Financial Cryptography '98*, volume 1465 of *Lecture Notes in Computer Science*, pages 178–183. Springer, 1998. doi:10.1007/BFb0055482.

40 Omri Shmueli. Semi-quantum tokenized signatures. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part I*, volume 13507 of *Lecture Notes in Computer Science*, pages 296–319, Santa Barbara, CA, USA, August 15–18 2022. Springer, Heidelberg, Germany. `doi:10.1007/978-3-031-15802-5_11`.

41 S. Stubblebine. Recent-secure authentication: enforcing revocation in distributed systems. In *2012 IEEE Symposium on Security and Privacy*, page 0224, Los Alamitos, CA, USA, May 1995. IEEE Computer Society.

42 Dominique Unruh. Revocable quantum timed-release encryption. *J. ACM*, 62(6):49:1–49:76, 2015.

43 Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983.

44 Mark Zhandry. Quantum lightning never strikes the same state twice. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019, Part III*, volume 11478 of *Lecture Notes in Computer Science*, pages 408–438, Darmstadt, Germany, May 19–23 2019. Springer, Heidelberg, Germany. `doi:10.1007/978-3-030-17659-4_14`.

The Quantum Decoding Problem

André Chailloux
Inria de Paris, France

Jean-Pierre Tillich
Inria de Paris, France

Abstract

One of the founding results of lattice based cryptography is a quantum reduction from the Short Integer Solution (SIS) problem to the Learning with Errors (LWE) problem introduced by Regev. It has recently been pointed out by Chen, Liu and Zhandry [12] that this reduction can be made more powerful by replacing the LWE problem with a quantum equivalent, where the errors are given in quantum superposition. In parallel, Regev's reduction has recently been adapted in the context of code-based cryptography by Debris, Remaud and Tillich [14], who showed a reduction between the Short Codeword Problem and the Decoding Problem (the DRT reduction). This motivates the study of the Quantum Decoding Problem (QDP), which is the Decoding Problem but with errors in quantum superposition and see how it behaves in the DRT reduction.

The purpose of this paper is to introduce and to lay a firm foundation for QDP. We first show QDP is likely to be easier than classical decoding, by proving that it can be solved in *quantum polynomial time* in a large regime of noise whereas no non-exponential quantum algorithm is known for the classical decoding problem. Then, we show that QDP can even be solved (albeit not necessarily efficiently) beyond the information theoretic Shannon limit for classical decoding. We give precisely the largest noise level where we can solve QDP giving in a sense the information theoretic limit for this new problem. Finally, we study how QDP can be used in the DRT reduction. First, we show that our algorithms can be properly used in the DRT reduction showing that our quantum algorithms for QDP beyond Shannon capacity can be used to find minimal weight codewords in a random code. On the negative side, we show that the DRT reduction cannot be, in all generality, a reduction between finding small codewords and QDP by exhibiting quantum algorithms for QDP where this reduction entirely fails. Our proof techniques include the use of specific quantum measurements, such as q-ary unambiguous state discrimination and pretty good measurements as well as strong concentration bounds on weight distribution of random shifted dual codes, which we relate using quantum Fourier analysis.

2012 ACM Subject Classification Theory of computation → Quantum information theory; Theory of computation → Error-correcting codes; Security and privacy → Cryptanalysis and other attacks

Keywords and phrases quantum information theory, code-based cryptography, quantum algorithms

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.6

Related Version *Full Version*: `https://arxiv.org/pdf/2310.20651`

1 General cryptographic context

Error correcting codes appeared first as the fundamental tool to transmit information reliably through a noisy channel [25] and has found numerous applications in information theory and complexity. The hardness - even for quantum computers - of decoding random linear codes is also the core of code-based cryptography. In the cryptographic context, the decoding problem corresponds to decoding the k-dimensional vector space $\mathscr{C}$ (*i.e.*, the code) generated by the rows of a randomly generated $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ (which is called a *generating matrix* of the code):

$$\mathscr{C} \stackrel{\triangle}{=} \left\{ \boldsymbol{u}\mathbf{G} \colon \boldsymbol{u} \in \mathbb{F}_q^k \right\} . \tag{1}$$

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 6; pp. 6:1–6:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Here $\mathbb{F}_q$ denotes the finite field with q elements. In the decoding problem, we are given the noisy codeword $\boldsymbol{c}+\boldsymbol{e}$ where $\boldsymbol{c}$ belongs to $\mathscr{C}$ and we are asked to find the original codeword $\boldsymbol{c}$.

▶ **Problem 1** (DP(q,n,k,f)). *The decoding problem with positive integer parameters q,n,k and a probability distribution f on $\mathbb{F}_q^n$ is defined as:*

- *Input: $(\mathbf{G},\boldsymbol{c}+\boldsymbol{e})$ where $\mathbf{G}\in\mathbb{F}_q^{k\times n}$ and $\boldsymbol{u}\in\mathbb{F}_q^k$ are sampled uniformly at random over their domain - which generates a random codeword $\boldsymbol{c}=\boldsymbol{u}\mathbf{G}$ - and $\boldsymbol{e}$ is sampled from the distribution f.*
- *Goal: from $(\mathbf{G},\boldsymbol{c}+\boldsymbol{e})$, find $\boldsymbol{c}$.*

This problem for random codes has been studied for a long time and despite many efforts on this issue, the best (even quantum) algorithms are exponential in the codelength n for natural noise distributions f in the regime where k is linear in n and the rate $R\stackrel{\triangle}{=}\frac{k}{n}$ bounded away from 0 and 1 [23, 27, 15, 19, 5, 20, 18, 9, 8]. This remains true even if we consider quantum algorithms

The most common noise distribution studied in this context is the uniform distribution over the errors of fixed Hamming weight t, but there are also other distributions, like in the binary case ($q=2$) the i.i.d Bernoulli distribution model which is frequently found in the Learning Parity with Noise problem (LPN) [16]. In code-based cryptography, the regime which is almost always relevant is a fixed number of samples n (or codelength) in the linear regime *i.e.* $k=\Theta(n)$. We will focus on this case here. While the security of many code-based cryptosystems relies on the hardness of the decoding problem, it can also be based on finding a “short” codeword (as in [21] or in [2, 7, 29] to build collision resistant hash functions), a problem which is stated as follows.

▶ **Problem 2** (SCP(q,n,k,w)). *The short codeword problem with parameters $q,n,k,w\in\mathbb{N}$ is defined as:*

- *Given: $\mathbf{H}\in\mathbb{F}_q^{(n-k)\times n}$ which is sampled uniformly at random,*
- *Find: $\boldsymbol{c}\in\mathbb{F}_q^n\setminus\{\mathbf{0}\}$ such that $\mathbf{H}\boldsymbol{c}^\intercal=\mathbf{0}$ and the weight $|\boldsymbol{c}|$ of $\boldsymbol{c}$ satisfies $|\boldsymbol{c}|\leq w$.*

Here we are looking for a non-zero codeword $\boldsymbol{c}$ of weight $\leq w$ in the k-dimensional code $\mathscr{C}$ defined by the so-called parity-check matrix $\mathbf{H}$, namely[1] :

$$\mathscr{C}\stackrel{\triangle}{=}\left\{\boldsymbol{c}\in\mathbb{F}_q^n:\mathbf{H}\boldsymbol{c}^\intercal=\vec{0}\right\}.$$

The weight function which is generally used here is the Hamming weight, *i.e.* for a vector $\boldsymbol{x}=(x_1,\cdots,x_n)\in\mathbb{F}_q^n$, its Hamming weight is defined as

$$|\boldsymbol{x}|\stackrel{\triangle}{=}\#\{i\in[\![1,n]\!]:x_i\neq 0\}.$$

We will only deal with this weight here. Decoding and looking for short codewords are problems that have been conjectured to be extremely close. They have been studied for a long time, and the best algorithms for solving these two problems are the same, namely Information Set Decoding algorithms [23, 27, 15, 5, 20, 6].

Recently, Debris-Alazard, Remaud and Tillich showed a quantum reduction from SCP to DP adapting Regev’s reduction from the Short Integer Solution (SIS) problem to the Learning With Errors problem (LWE). It has recently been pointed out by Chen, Liu and

[1] The short codeword problem is usually defined by picking a random parity-check matrix $\mathbf{H}\in\mathbb{F}_q^{(n-k)\times n}$ and not a random generating matrix $\mathbf{G}\in\mathbb{F}_q^{k\times n}$ but the differences are minor (see for example [13]) and one could also define this problem via the generating matrix of a code as we did for the decoding problem.

Zhandry [12] that Regev's reduction can be made more powerful by replacing the LWE problem with a quantum equivalent, where the errors are given in quantum superposition. It is therefore natural to ask whether having errors in quantum superposition in the decoding problem can be applied to the DRT reduction in order to improve it.

The purpose of this article is to introduce and to lay a firm foundation for the Quantum Decoding Problem, which is the decoding problem but with errors in quantum superposition. We first present the DRT reduction for codes, properly define QDP, and then present in detail our contributions.

2 Regev's quantum reduction and follow-up work

Regev's quantum reduction[24] is at the core of complexity reductions for these problems, which with [1] essentially started lattice-based cryptography. His approach when rephrased in the coding context is based on the following observation. Suppose that we were able to construct a quantum superposition $\sqrt{\frac{1}{Z}}\sum_{\boldsymbol{c}\in\mathcal{C}}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(e)}|\boldsymbol{c}+\boldsymbol{e}\rangle$ of noisy codewords of some code $\mathcal{C}$ over $\mathbb{F}_q$, for a normalization factor Z. By applying the quantum Fourier transform on such a state, because of the periodicity property of such a state, we get a superposition concentrating solely on the codewords of the dual $\mathcal{C}^\perp$ of $\mathcal{C}$, that is $\frac{1}{\sqrt{Z}}\sum_{\boldsymbol{c}^\perp\in\mathcal{C}^\perp}\sqrt{\widehat{f}(\boldsymbol{c}^\perp)}|\boldsymbol{c}^\perp\rangle$. Here $\widehat{f}$ is the (classical) Fourier transform of f. Recall that the dual code is defined as

▶ **Definition 1** (dual code). *Let $\mathcal{C}$ be a k-dimensional linear code over $\mathbb{F}_q$ of length n. Let $\boldsymbol{x}\cdot\boldsymbol{y}$ be the inner product of vectors $\boldsymbol{x}$, $\boldsymbol{y}$ in $\mathbb{F}_q^n$ defined as $\boldsymbol{x}\cdot\boldsymbol{y}=\sum_i x_iy_i$. The dual code $\mathcal{C}^\perp$ is an $(n-k)$ dimensional subspace of $\mathbb{F}_q^n$ defined by $\mathcal{C}^\perp\stackrel{\triangle}{=}\{\boldsymbol{d}\in\mathbb{F}_q^n:\boldsymbol{d}\cdot\boldsymbol{c}=0,\ \forall\boldsymbol{c}\in\mathcal{C}\}$*

We can expect that if f concentrates on fairly small weights, then $\widehat{f}$ also concentrates on small weights. This gives a way of sampling low weight (dual) codewords and solve SCP for the dual code. The point is now that $\sqrt{\frac{1}{Z}}\sum_{\boldsymbol{c}\in\mathcal{C}}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(e)}|\boldsymbol{c}+\boldsymbol{e}\rangle$ can be obtained by solving DP on states that are easy to construct. This is the main idea of the DRT reduction. More precisely, the whole algorithm works as

Step 1. Creation of a superposition of noise tensored with a uniform superposition of codewords

$$|\phi_1\rangle=\sqrt{\frac{1}{q^k}}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(e)}|\boldsymbol{e}\rangle\sum_{\boldsymbol{c}\in\mathcal{C}}|\boldsymbol{c}\rangle.$$

Step 2. Entangling the codeword with the noise by adding the second register to the first one

$$|\phi_2\rangle=\sqrt{\frac{1}{q^k}}\sum_{\boldsymbol{c}\in\mathcal{C}}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(e)}|\boldsymbol{c}+\boldsymbol{e}\rangle|\boldsymbol{c}\rangle.$$

Step 3. Disentangling the two registers by decoding $\boldsymbol{c}+\boldsymbol{e}$ and therefore finding $\boldsymbol{c}$ which allows to erase the second register (a different normalization Z arises when decoding is imperfect and we condition on measuring $\mathbf{0}$ in the last register).

$$|\phi_3\rangle=\sqrt{\frac{1}{Z}}\sum_{\boldsymbol{c}\in\mathcal{C}}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(e)}|\boldsymbol{c}+\boldsymbol{e}\rangle|\mathbf{0}\rangle.$$

Step 4. Applying the quantum Fourier transform on the first register and get

$$\frac{1}{\sqrt{Z}}\sum_{\boldsymbol{d}\in\mathcal{C}^{\perp}}\sqrt{\widehat{f}(\boldsymbol{d})}|\boldsymbol{d}\rangle|\mathbf{0}\rangle$$

Step 5. Measure the first register and get some $\boldsymbol{d}$ in $\mathcal{C}^{\perp}$.

This approach is at the heart of the quantum reductions obtained in [24, 26, 14]. It is also a crucial ingredient in the paper [28] proving verifiable quantum advantage by constructing - among other things - one-way functions that are even collision resistant against classical adversaries but are easily invertible quantumly. In [24, 26, 14], the crucial erasing/disentangling step is performed with the help of a *classical* decoding algorithm. Indeed any (classical or quantum) algorithm that can recover $\boldsymbol{c}$ from $\boldsymbol{c}+\boldsymbol{e}$ can be applied coherently to erase the last register in step 3[2] .

A key insight observed in [12] is that it is actually enough to recover $|\boldsymbol{c}\rangle$ from the state $\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(\boldsymbol{e})}|\boldsymbol{c}+\boldsymbol{e}\rangle$ so we are given a superposition of all the noisy codewords $\boldsymbol{c}+\boldsymbol{e}$ and not a fixed one. This means we have to solve the following problem

▶ **Problem 3** (QDP(q,n,k,f)). *The quantum decoding problem with positive integer parameters q,n,k and a probability distribution f on $\mathbb{F}_q^n$ is defined as:*

- *Input: Take $\mathbf{G}\in\mathbb{F}_q^{k\times n}$ and $\boldsymbol{u}\in\mathbb{F}_q^k$ sampled uniformly at random over their domain. Let $\boldsymbol{c}=\boldsymbol{u}\mathbf{G}$ and $|\psi_{\boldsymbol{c}}\rangle\stackrel{\triangle}{=}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(e)}|\boldsymbol{c}+\boldsymbol{e}\rangle$. The (quantum) input to this problem is $(\mathbf{G},|\psi_{\boldsymbol{c}}\rangle)$.*
- *Goal: given $(\mathbf{G},|\psi_{\boldsymbol{c}}\rangle)$, find $\boldsymbol{c}$.*

It's not clear a priori whether this is helpful or not. If one measures the state $|\psi_{\boldsymbol{c}}\rangle$ then one recovers a noisy codeword and we are back to the classical decoding problem. However, in the context of lattices, [12] showed that this approach can lead to improvements. A final small remark on the motivation of the quantum decoding problem. We are not in the context of noise coming from a realistic quantum channel so we do not need our noise model to emulate real quantum noise (which would certainly not be a q-ary symmetric channel with the same phases). The motivation of this definition really comes from an algorithmic and complexity perspective, as well as a quantum information-theoretic perspective but not from a quantum error correcting perspective.

3 Contributions

Here we focus on the noise model which is relevant for the Hamming metric in SCP, namely the Bernoulli noise of parameter p. This means we consider the error function

$$f(\boldsymbol{e})=(1-p)^{n-|\boldsymbol{e}|}\left(\frac{p}{q-1}\right)^{|\boldsymbol{e}|}.$$

which in turn means that for any $\boldsymbol{c}=(c_1,\ldots,c_n)\in\mathbb{F}_q^n$, we can rewrite

$$|\psi_{\boldsymbol{c}}\rangle\stackrel{\triangle}{=}\sum_{\boldsymbol{e}\in\mathbb{F}_q^n}\sqrt{f(\boldsymbol{e})}|\boldsymbol{c}+\boldsymbol{e}\rangle=\bigotimes_{i=1}^{n}\left(\sqrt{1-p}|c_i\rangle+\sum_{\alpha\in\mathbb{F}_q^*}\sqrt{\frac{p}{q-1}}|c_i+\alpha\rangle\right).$$

[2] Indeed, having such an algorithm means we can construct the unitary $U:|\boldsymbol{c}+\boldsymbol{e}\rangle|0\rangle\rightarrow|\boldsymbol{c}+\boldsymbol{e}\rangle|\boldsymbol{c}\rangle$. Applying the inverse of this unitary will give the erasure operation.

For this Bernoulli noise with parameter p, the associated quantum decoding problem is written $\mathrm{QDP}(q,n,k,p)$. We will lay out a firm foundation for this quantum decoding problem by focusing on the case which is generally relevant in code-based cryptography, namely in the fixed code rate $R \stackrel{\triangle}{=} k/n$ regime and will show that QDP *departs significantly* from the classical decoding problem because we show that

(i) QDP is likely to be easier than classical decoding, by proving that it can be solved in *polynomial time* in a large regime of noise whereas no non-exponential algorithm is known for the classical decoding problem.

(ii) the problem can even be solved (albeit not necessarily efficiently) beyond the information theoretic Shannon limit for classical decoding. We will give precisely the largest noise level where we can solve QDP giving in a sense the information theoretic limit for the new problem.

(iii) We study how these QDP algorithms fit in the DRT reduction. We show using our quantum polynomial algorithm in this reduction in order to obtain quantum polynomial algorithms recovering Prange's bound. Even more interestingly, we show that our quantum algorithm for QDP of (ii) can be used in order to find small codewords of weight as small as the *minimal distance* of the code, which is the best we can hope for. On the negative side, we show that the DRT reduction cannot be, in all generality, a reduction between QDP and finding small codewords by exhibiting quantum algorithms for QDP that make this reduction entirely fail.

We now perform a detailed description of our contributions.

3.1 Polynomial time quantum algorithm for QDP in a large parameter regime

Our first result is the following

▶ **Theorem 2.** *Let $R \in [0,1]$. For any $p < \left(\frac{(q-1)R}{q}\right)^{\perp}$, there exists a quantum algorithm that solves* $\mathrm{QDP}(q,n,\lfloor Rn \rfloor,p)$ *wp.* $1-2^{-\Omega(n)}$ *in time* $\mathrm{poly}(n,\log(q))$*, where for a real number $x \in [0,1]$, $x^{\perp} = \frac{\left(\sqrt{(1-x)(q-1)}-\sqrt{x}\right)^2}{q}$.*

Let us dive in how we obtain this result. We start from an input of $\mathrm{QDP}(q,n,\lfloor Rn \rfloor,p)$, which, for an (unknown) codeword $\boldsymbol{c} = c_1,\dots,c_n$ is the state

$$|\Psi_{\boldsymbol{c}}\rangle = \bigotimes_{i=1}^{n} |\psi_{c_i}\rangle \quad \text{with} \quad |\psi_{c_i}\rangle \stackrel{\triangle}{=} \sqrt{1-p}|c_i\rangle + \sum_{\alpha\in\mathbb{F}_q^*} \sqrt{\frac{p}{q-1}}|c_i+\alpha\rangle$$

and the goal is to recover $\boldsymbol{c}$. Our algorithm performs Unambiguous State Discrimination (USD) on each of the n registers. USD is a quantum measurement that, on input $|\psi_{c_i}\rangle$, will output or the correct value c_i, or an abort symbol $\perp$ but will never output a value $\alpha \in \mathbb{F}_q$ different from c_i. Then, if we have enough correct values of c_i (essentially more than $\lceil Rn \rceil$), then one can recover the whole $\boldsymbol{c}$ using the description of the code and basic linear algebra.

Optimal unambiguous state discrimination is very well understood in the binary case ($q=2$) but is not known in general for more than 2 states. In certain situations where we have a symmetric set of states [10] we know how to perform optimal USD. This would apply in our case case where q is prime. We have generalized the approach of [10] to be able to apply it to any finite field size q, and prove the following.

▶ **Proposition 3.** *Let q be a prime power, and $f : \mathbb{F}_q \to \mathbb{C}$ st. $\|f\|_2 = 1$. For each $y \in \mathbb{F}_q$, we define $|\psi_y\rangle = \sum_{\alpha \in \mathbb{F}_q} f(\alpha)|y+\alpha\rangle$. There exists a quantum measurement that, when given $|\psi_y\rangle$, outputs y wp. p_{USD} and $\perp$ wp. $1-p_{USD}$ where $p_{USD} = q \cdot \min_{\alpha \in \mathbb{F}_q} |\widehat{f}(\alpha)|^2$, and this is optimal. Moreover, if f corresponds to a Bernoulli noise of parameter p, this measurement can be done in time* $\mathrm{polylog}(q)$ *and we have $p_{USD} = \frac{qp^\perp}{q-1}$, where $p^\perp = \frac{\left(\sqrt{(1-p)(q-1)} - \sqrt{p}\right)^2}{q}$.*

Notice that [12] proposed a USD measurement with $p_{USD} = \frac{1}{q} \min_{\alpha \in \mathbb{F}_q} |\widehat{f}(\alpha)|^2$. Their measurement does not scale well with q contrarily to our measurement which has basically the right scaling with q. For instance, [12] requires that $q = \mathrm{poly}(n)$. We have no such restriction in our case and our algorithms work in polynomial time even for $q = 2^{\Omega(n)}$.

3.1.1 Interpretation as changing the noise channel

A nice interpretation of the above algorithm is that when the error is in quantum superposition, one can use quantum measurements to change the noise model. For instance, when we are given $|\psi_{c_i}\rangle = \sqrt{1-p}|c_i\rangle + \sum_{\alpha \in \mathbb{F}_q^*} \sqrt{\frac{p}{q-1}}|c_i + \alpha\rangle$ then

- One can measure in the computational basis to obtain c_i that has been flipped wp. p to one of the other $q-1$ values at random.
- One can use unambiguous state discrimination in which case c_i has been erased wp. $1 - \frac{qp^\perp}{q-1}$.

What we show in Theorem 2 is that the second strategy is actually much more powerful for recovering the codeword $\boldsymbol{c}$. A natural question to ask is whether this can further be generalized to other measurements. In the binary setting we actually generalize USD as follows: given $|\psi_{c_i}\rangle$, the measurement sometimes outputs $\perp$ but it can also fail with some small probability. We prove the following

▶ **Proposition 4** (Partial USD). *Let $p, s \in [0, \frac{1}{2})$ with $s \leq p$ and let $u = \frac{p^\perp}{s^\perp}$. There exists a quantum measurement that when applied to $|\psi_{c_i}\rangle = \sqrt{1-p}|c_i\rangle + \sqrt{p}|1-c_i\rangle$ outputs c_i wp. $u(1-s)$, $(1-c_i)$ wp. us and $\perp$ wp. $1-u$.*

Notice that this generalizes both the computational basis measurement (by taking $s = p$) and unambiguous state discrimination (by taking $s = 0$ giving $u = 2p^\perp$). This seems a very natural way of generalizing USD but is not something we have found in the literature and could be of independent interest. We can use this measurement not to provide new polynomial time algorithms but rather to give a reduction between different Quantum Decoding problems, which we detail in the full text.

3.2 Determining exactly the tractability of the quantum decoding problem

We are now interested in the tractability of $\mathrm{QDP}(q, n, k, p)$ meaning when is it possible from an information theoretic perspective to solve this problem. A fundamental quantity is relavant here, namely the Gilbert-Varshamov distance $\delta_{\min}(R)$ defined below

▶ **Notation 1.** *Let $R \in [0,1]$. We define $\delta_{\min}(R) \triangleq h_q^{-1}(1-R)$, where $h_q(x) \triangleq -(1-x)\log_q(1-x) - x\log_q\left(\frac{x}{q-1}\right)$. h_q is a bijection from $x \in \left[0, \frac{q-1}{q}\right]$ to $[0,1]$ and $h_q^{-1} : [0,1] \to \left[0, \frac{q-1}{q}\right]$ is the inverse of h_q.*

For the classical setting, it is well understood that $\mathrm{DP}(q,n,k,p)$ is not tractable when $p > \delta_{\min}(\frac{k}{n})$, meaning that even an unbounded algorithm will solve the problem wp. $o(1)$. We would like now to understand what happens in the quantum setting. Techniques based on (partial) USD will not work in the regime $p > \delta_{\min}(R)$. Since we are only interested in the tractability of the problem, we can consider optimal quantum algorithms for discriminating between the states $|\Psi_{\boldsymbol{c}}\rangle = \sum_{\boldsymbol{e}\in\mathbb{F}_q^n} \sqrt{f(\boldsymbol{e})}|\boldsymbol{c}+\boldsymbol{e}\rangle$ where f accounts for the Bernoulli noise of parameter p. This problem can be addressed by using the *Pretty Good Measurement* (PGM) which has turned out to be a very useful tool in quantum information. If we define $\mathrm{P}_{\mathrm{PGM}}$ as the probability that the pretty good measurement succeeds in solving our problem and define $\mathrm{P}_{\mathrm{OPT}}$ as the maximal probability that any measurement succeeds, we have [4, 22]

$$\mathrm{P}_{\mathrm{OPT}}^2 \leq \mathrm{P}_{\mathrm{PGM}} \leq \mathrm{P}_{\mathrm{OPT}}.$$

This means that in order to study the tractability of the quantum decoding problem, it is enough to look at the PGM associated with the problem of distinguishing the states $\{|\Psi_{\boldsymbol{c}}\rangle\}$. We show that

▶ **Theorem 5.** *Let $R \in (0,1)$.*

- *For $p < (\delta_{\min}(1-R))^{\perp}$, $\mathrm{QDP}(q,n,\lfloor Rn \rfloor,p)$ can be solved using the PGM wp. $P_{PGM} = 1-o(1)$ hence the problem is tractable.*
- *For $p > (\delta_{\min}(1-R))^{\perp}$, the probability that the PGM solves this problem is $P_{PGM} = o(1)$ hence $P_{OPT} = o(1)$ and the problem is intractable.*

In order to prove this theorem, we study the Pretty Good Measurement associated to the states $|\Psi_{\boldsymbol{c}}\rangle$ which are the possible inputs of QDP. In order to study our PGM, we define the shifted dual codes of $\mathcal{C}$

$$\mathcal{C}_{\boldsymbol{s}}^{\perp} \stackrel{\triangle}{=} \{\boldsymbol{x} \in \mathbb{F}_q^n : \mathbf{G}\boldsymbol{x}^{\intercal} = \boldsymbol{s}\}$$

We show the following

▶ **Proposition 6.** *We consider the Pretty Good Measurement associated to the states $\{|\Psi_{\boldsymbol{c}}\rangle\}_{\boldsymbol{c}\in\mathcal{C}}$ with $|\Psi_{\boldsymbol{c}}\rangle = \sum_{\boldsymbol{e}} f(\boldsymbol{e})|\boldsymbol{c}+\boldsymbol{e}\rangle$. This measurement outputs $\boldsymbol{c}$ given $|\Psi_{\boldsymbol{c}}\rangle$ wp.*

$$p_{PGM} = \frac{1}{q^k}\left(\sum_{\boldsymbol{s}\in\mathbb{F}_q^k} n_{\boldsymbol{s}}\right)^2 \quad \text{where} \quad n_{\boldsymbol{s}} = \sqrt{\sum_{\boldsymbol{y}\in\mathcal{C}_{\boldsymbol{s}}^{\perp}} |\hat{f}(\boldsymbol{y})|^2}.$$

This shows an interesting and unexpected link between the Pretty Good Measurement associated to the states $\{|\Psi_{\boldsymbol{c}}\rangle\}$ and the shifted dual codes $\mathcal{C}_{\boldsymbol{s}}^{\perp}$. In the particular case of a Bernoulli noise of parameter p, the value of $n_{\boldsymbol{s}}$ will be dominated by a quantity related to the number of words of weight close to $p^{\perp}$ in the shifted dual code $\mathcal{C}_{\boldsymbol{s}}^{\perp}$. In order to conclude, we use strong concentration bounds on the weight distribution of shifted dual codes.

3.2.1 Comparing the complexity of DP and QDP

With this full characterization, we compare the hardness, and tractability of the classical and quantum decoding problems. For $p = 0$, we have of course a polynomial time algorithm to solve $\mathrm{DP}(q,n,\lfloor Rn \rfloor,0)$. For $0 < p \leq \delta_{\min}(R)$, the problem is tractable and the best known classical or quantum algorithms run in time $2^{\Omega(n)}$. For $p > \delta_{\min}(R)$, we know the problem is intractable. For the Quantum Decoding Problem, we obtain a very different picture. A comparison of these results is presented in Figures 1 and 2 where we use the following terminology

- Easy: there exists an algorithm that runs in time poly(n).
- Hard: the best known (classical or quantum) algorithm runs in time $2^{\Omega(n)}$, but there could potentially be more efficient algorithms.
- Intractable: we know that any (even unbounded) algorithm can solve the problem wp. at most $o(1)$.

Figure 1 Hardness and tractability of the decoding problem $\mathrm{DP}(q, n, \lfloor Rn \rfloor, p)$, for any fixed $R \in [0, 1]$, as a function of p.

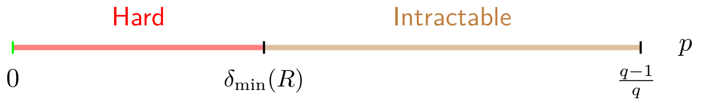
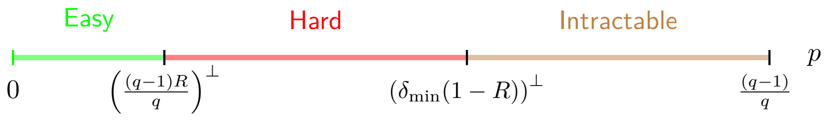


Figure 2 Hardness and tractability of the quantum decoding problem $\mathrm{QDP}(q, n, \lfloor Rn \rfloor, p)$, for any fixed $R \in [0, 1]$, as a function of p.

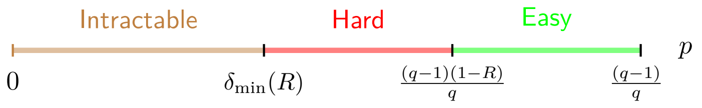

Easy
Hard
Intractable
p
0
$\left(\frac{(q-1)R}{q}\right)^{\perp}$
$(\delta_{\min}(1-R))^{\perp}$
$\frac{(q-1)}{q}$

This gives a proper characterization of the difficulty of QDP. In our next contribution, we will apply them in the DRT quantum reduction in order to derive some results for the short codeword problem. As we will show, the results from Figure 2 will match exactly our knowledge for the short codeword problem.

3.3 Using our algorithms in Regev's reduction

We are now interested in solving the short codeword problem using Regev's reduction and the algorithms we described in the previous section. The known (classical and quantum) hardness of the short codeword problem is summarized in Fig. 3. In our coding context, the only known reduction is the following

Figure 3 Hardness and tractability of the short codeword problem $\mathrm{SCP}(q, n, \lfloor Rn \rfloor, p)$ for a fixed $R \in (0, 1)$, as a function of p.



▶ **Proposition 7** ([14], informal). *Fix integers $n, q \geq 2$ as well as parameters $R, p \in (0, 1)$ st. $p \leq \delta_{\min}(R)$. From any quantum algorithm that solves $DP(q, n, \lceil (1-R)n \rceil, p)$ with high probability, there exists a quantum algorithm that solves $SCP(q, n, \lfloor Rn \rfloor, p^{\perp})$ with high probability where recall that $p^{\perp} = \frac{\left(\sqrt{(1-p)(q-1)} - \sqrt{p}\right)^2}{q}$.*

In some sense, this reduction is far from tight. Indeed, if we take the best known algorithms for DP (see Figure 1) and we apply the above proposition, we get quantum algorithms much worst than the best known ones from Figure 3. On the other hand, if we could plug in our algorithms for QDP in this reduction, we would obtain quantum algorithms for SCP that have the same complexities as the one in Figure 3. From our discussion in Section 2, it seems that one could perform the same reduction as above but replacing DP with QDP. The reality is quite more tricky. Indeed, if the quantum algorithm for QDP succeeds wp. 1 then the reduction works. However, even a small error in the quantum algorithm for QDP can lead to large error in the corresponding algorithms for SCP.

We first show that this is not an issue when we use the quantum algorithms for QDP we described in the previous section in the DRT reduction.

▶ **Theorem 8.** *For any $p < \left(\frac{(q-1)R}{q}\right)^{\perp}$, if we plug the quantum polynomial time algorithm of Theorem 2 for* $\mathrm{QDP}(q, n, \lfloor(1-R)n\rfloor, p^{\perp})$ *in Regev's reduction, we obtain a quantum polynomial time algorithm for SCP$(q, n, \lfloor R \rfloor, p)$.*

▶ **Theorem 9.** *For any $p < \delta_{min}(1-R)$, if we plug (a slight variant of) the quantum algorithm of Theorem 5 for* $\mathrm{QDP}(q, n, \lfloor(1-R)n\rfloor, p^{\perp})$ *in Regev's reduction, we obtain a quantum polynomial time algorithm for SCP$(q, n, \lfloor R \rfloor, p)$.*

3.3.1 Efficiency of the reduction

We graphically compare here the DRT reduction using DP and using our algoirthms for QDP.

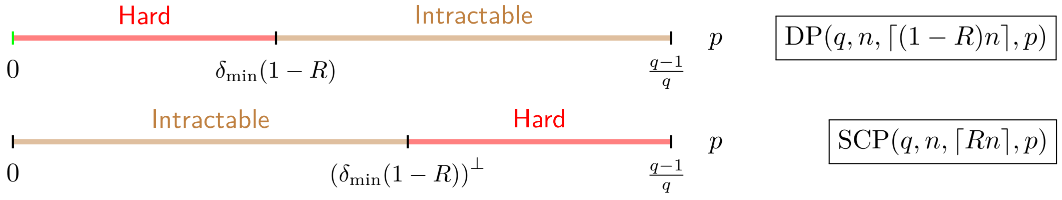

Hard
Intractable
0
$\delta_{\min}(1-R)$
$\frac{q-1}{q}$
p
$\mathrm{DP}(q, n, \lceil(1-R)n\rceil, p)$
Intractable
Hard
0
$(\delta_{\min}(1-R))^{\perp}$
$\frac{q-1}{q}$
p
$\mathrm{SCP}(q, n, \lceil Rn\rceil, p)$

Figure 4 On the top, best known (classical or quantum) algorithms for $\mathrm{DP}(q, n, \lceil(1-R)n\rceil, p)$. On the bottom, complexity of a quantum algorithm for $\mathrm{SCP}(q, n, \lceil Rn\rceil, p)$ that uses the best algorithm for $\mathrm{DP}(q, n, \lceil(1-R)n\rceil, p)$ and then uses Proposition 7.

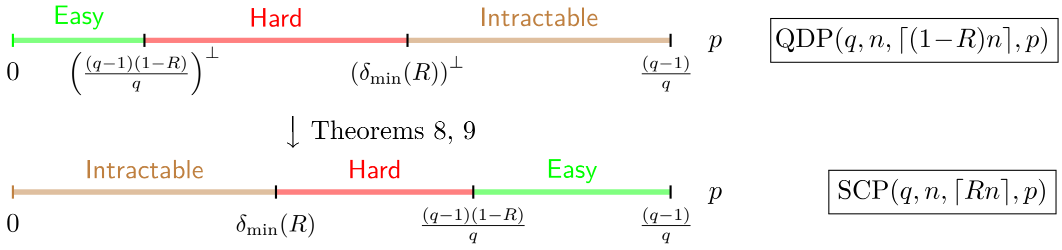

Easy
Hard
Intractable
0
$\left(\frac{(q-1)(1-R)}{q}\right)^{\perp}$
$(\delta_{\min}(R))^{\perp}$
$\frac{(q-1)}{q}$
p
$\mathrm{QDP}(q, n, \lceil(1-R)n\rceil, p)$
Theorems 8, 9
Intractable
Hard
Easy
0
$\delta_{\min}(R)$
$\frac{(q-1)(1-R)}{q}$
$\frac{(q-1)}{q}$
p
$\mathrm{SCP}(q, n, \lceil Rn\rceil, p)$

Figure 5 On the top, our quantum algorithms for $\mathrm{QDP}(q, n, \lceil(1-R)n\rceil, p)$. On the bottom, complexity of a quantum algorithm for $\mathrm{SCP}(q, n, \lceil Rn\rceil, p)$ that would use our algorithms $\mathrm{QDP}(q, n, \lceil(1-R)n\rceil, p)$ and then Theorems 8 and 9.

What we find quite remarkable is that our algorithm for QDP used at the limit of the tractability bound can be used to recover minimal weight codewords of weight $\delta_{\min}(R)$ in the dual. A natural question is whether we can have generic reductions between SCP and QDP. We show that the DRT reduction fails for this task and the increase in error in the reduction can be drastic.

▶ **Theorem 10.** *For any $p < \delta_{min}(1-R)$, there exists a quantum algorithm that solves* $\mathrm{QDP}(q, n, \lfloor(1-R)n\rfloor, p^{\perp})$ *wp.* $1 - o(1)$ *st. if we plug it in the DRT reduction, the resulting algorithm for SCP*$(q, n, \lfloor Rn\rfloor, p)$ *never succeeds.*

These results show that, while it is impossible to have a generic reduction from SCP to QDP with this method, it is (at least in our examples) possible to find algorithms for QDP that give results according to Fig. 5, and recover the areas where the problem is easy or tractable. This can be seen as quite a surprise since our bounds on QDP come from information theory and best known bounds on SCP comes from classical coding theory and seem unrelated at first.

4 Related work

Our main starting point is [12] so it natural to compare our contributions with this work. In [12], they introduce the S-$|\mathrm{LWE}\rangle$ problem, the lattice equivalent of QDP. They construct from it a quantum algorithm for SIS_{∞} via Regev's reduction while our work focuses on QDP mainly for its own sake. Regarding their quantum polynomial time algorithm for S-$|\mathrm{LWE}\rangle$, it is obtained by performing a variant of Unambiguous State discrimination where we only rule out certain values for the code-symbols, and then they use the Arora-Ge algorithm [3] for recovering completely the codeword by solving an algebraic system which for the parameters that are considered there, is of polynomial complexity. Our quantum polynomial time algorithm is inspired by this approach but since we work with the q-ary Bernoulli noise, we can directly use unambiguous state discrimination. Also, we perform a more efficient q-ary USD which allows us to work even when $q = 2^{\Omega(n)}$ while the work of [12] works only when $q = \mathrm{poly}(n)$. Our other results on the (in)tractability results, as well as the discussion on the DRT reduction are entirely novel and do not have an equivalent version in [12].

Another quantum variant has been presented [17] but where they consider a quantum superposition of samples *i.e.* superpositions of generating matrices. They show that in this case the problem can be solved in quantum polynomial time. Their setting is very different from ours as we fix a code and do not have codes in superposition. Moreover, their techniques are not applicable to our setting.

A recent result [11] also studies the S-$|\mathrm{LWE}\rangle$ problem. They perform a quantum reduction between LWE and S-$|\mathrm{LWE}\rangle$ with extra unknown phases. The parameters of this S-$|\mathrm{LWE}\rangle$ are such that if they did not have these unknown phases, they could solve it with a subexponential quantum algorithm using a subexponential number of samples. This is very far from our parameter range and hence not comparable but gives an interesting use of Kuperberg's sieve for this kind of problem.

5 Discussion

5.1 A problem which is interesting in its own

Our work lays firm theoretical foundations for the Quantum Decoding Problem, where we find an interesting parameter range where the problem can be solved in quantum polynomial time. Moreover, we precisely characterize up to what level of noise the problem is tractable from an information theoretic point of view. Finally we show how our algorithms can be used in Regev's reduction for finding short codewords.

Beyond this, it seems to us that the quantum decoding problem is a natural and important problem in its own. We did not study the quantum decoding problem to relate it to the classical decoding problem so the aim of our results is not to say something about the complexity of the classical decoding problem (even though it is linked to the related Short Codeword Problem via Regev's reduction). We are actually more interested in the differences between these two problems. Having errors in quantum superposition can be used to change the noise model from a q-ary symmetric channel to an erasure channel. We even have a complete characterization in the binary setting via our notion of partial unambiguous state discrimination. Also quite surprisingly, we can decode beyond classical information theoretic limits. Moreover, the optimal bound $(\delta_{\min}(1-R))^{\perp}$ that we exhibit in the binary setting is exactly the first linear programming which bounds the minimal distance of a code depending on its size so this bound, which comes from the study of the Pretty Good Measurement has links, again, with fundamental quantities from classical coding theory.

But as we said, we have another strong motivation for studying the quantum decoding problem. Indeed, it is the problem which directly appears when performing Regev's reduction. This means studying the quantum decoding problem also gives us a better understanding of this reduction both from a complexity point of view and from a quantum algorithmic point of view. From a complexity point of view, our results explain why this reduction between the Decoding Problem and the Short Codeword problem (or equivalently between LWE and SIS) gives far from tight results. It is because this reduction is genuinely a reduction between the Quantum Decoding Problem and the Short Codeword and our analysis shows the former is much simpler than its classical counterpart. Also, Figure 5 shows that with this reduction, we recover the polynomial zone and the tractability zone of the short codeword problem which shows in some sense the tightness of this reduction. The fact that bounds on optimal Unambiguous State Discrimination and on the Pretty Good Measurement leads via Regev's reduction to Prange's bound (for the polynomial case) and to the Gilbert-Varshamov bound (for the tractability bound) shows interesting and, in our opinion, quite aesthetic links between quantum and classical information theory.

From an algorithmic point of view, these results can be seen as a new class of quantum algorithms for the short codeword problem. One might say that we do not improve on existing algorithms. For example, our quantum polynomial algorithm finds short codeword for some weights t that can also be found by the classical Prange's algorithm. Let us just observe here that if we could find a polynomial quantum algorithm that would beat Prange's bound then that would significantly change our understanding of the quantum hardness of these problems and the post-quantum security claims of code-based cryptography (and may be even lattice-based cryptography) would be affected. In the exponential regime, we propose a new family of quantum algorithms for the short codeword problem and there are many directions (changing the noise function f, determining the complexity of measurements required in the QDP, strict analysis in Regev's reduction ...) that look very promising for future work.

5.2 Technical takeaways

In the first part of the paper, we use binary Unambiguous State Discrimination for constructing our quantum polynomial algorithm. Once the idea is found, the techniques used are known and simple. We then extend this to partial unambiguous state discrimination - where we still allow some probability of failure much less than in Helstrom's measurement. This is a technique that we have not seen previously in the literature and could be of independent interest. In the q-ary setting, we extend optimal bounds for unambiguous state discrimination of symmetric states to the case q is a prime power and also show how to construct this measurement in time $\log(q)$ for Bernoulli noise. The second part of the paper, which deals with (in)tractability bounds is arguably the most technical part of the paper. A first interesting technical result was to precisely characterize the Pretty Good Measurement used in the quantum decoding problem of a code $\mathcal{C}$ as a projective measurement that involves the shifted dual codes of $\mathcal{C}$. Then, the most technical part was to actually compute the success probability of the Pretty Good Measurement as a function of the noise rate which requires precise and well used concentration and anti-concentration bounds on the weight distribution of shifted dual codes of a random code. In the third part of the paper, where we apply our algorithms to Regev's reduction, we mainly use our analysis of the Pretty Good Measurement developed in the previous section. Regarding the reduction using Unambiguous State Discrimination, the analysis is fairly simple. One interesting fact though is that we *do not* construct the state $\sum_{\boldsymbol{c},\boldsymbol{e}} f(\boldsymbol{e})|\boldsymbol{c}+\boldsymbol{e}\rangle$ but another state related to a punctured code $\mathcal{C}_J$ which allows us to find small dual codewords. This circumvents many issues arising when one wants to go from solving QDP (aka S-$|$LWE$\rangle$) to construct the state $\sum_{\boldsymbol{c},\boldsymbol{e}} f(\boldsymbol{e})|\boldsymbol{c}+\boldsymbol{e}\rangle$ (aka C-$|$LWE$\rangle$).

6 Proofs

The proofs of this article are presented in the full version of this paper (`https://arxiv.org/pdf/2310.20651`), which we omit here due to space restrictions.

References

1 Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In Gary L. Miller, editor, *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 99–108. ACM, 1996. `doi:10.1145/237814.237838`.

2 Benny Applebaum, Naama Haramaty, Yuval Ishai, Eyal Kushilevitz, and Vinod Vaikuntanathan. Low-complexity cryptographic hash functions. In *ITCS*, volume 67 of *LIPIcs*, pages 7:1–7:31. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017.

3 Sanjeev Arora and Rong Ge. New algorithms for learning in presence of errors. In Luca Aceto, Monika Henzinger, and Jiří Sgall, editors, *Automata, Languages and Programming*, volume 6755 of *LNCS*, pages 403–415. Springer Berlin Heidelberg, 2011. `doi:10.1007/978-3-642-22006-7_34`.

4 H. Barnum and E. Knill. Reversing quantum dynamics with near-optimal quantum and classical fidelity. *Journal of Mathematical Physics*, 43(5):2097–2106, April 2002. `doi:10.1063/1.1459754`.

5 Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding random binary linear codes in $2^{n/20}$: How $1+1=0$ improves information set decoding. In *Advances in Cryptology – EUROCRYPT 2012*, LNCS. Springer, 2012.

6 Leif Both and Alexander May. Optimizing BJMM with Nearest Neighbors: Full Decoding in $2^{2/21n}$ and McEliece Security. In *WCC Workshop on Coding and Cryptography*, September 2017. URL: `http://wcc2017.suai.ru/Proceedings{_}WCC2017.zip`.

7 Zvika Brakerski, Vadim Lyubashevsky, Vinod Vaikuntanathan, and Daniel Wichs. Worst-case hardness for LPN and cryptographic hashing via code smoothing. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III*, volume 11478 of *LNCS*, pages 619–635. Springer, 2019. `doi:10.1007/978-3-030-17659-4_21`.

8 Kevin Carrier, Thomas Debris-Alazard, Charles Meyer-Hilfiger, and Jean-Pierre Tillich. Statistical decoding 2.0: Reducing decoding to LPN. In *Advances in Cryptology - ASIACRYPT 2022*, LNCS. Springer, 2022. URL: `https://eprint.iacr.org/2022/1000`.

9 André Chailloux, Thomas Debris-Alazard, and Simona Etinski. Classical and quantum algorithms for generic syndrome decoding problems and applications to the lee metric. In Jung Hee Cheon and Jean-Pierre Tillich, editors, *Post-Quantum Cryptography*, pages 44–62, Cham, 2021. Springer International Publishing.

10 Anthony Chefles and Stephen M. Barnett. Optimum unambiguous discrimination between linearly independent symmetric states. *Physics Letters A*, 250(4):223–229, 1998. doi:10.1016/S0375-9601(98)00827-5.

11 Yilei Chen, Zihan Hu, Qipeng Liu, Han Luo, and Yaxin Tu. On the hardness of $\mathsf{S}|\mathsf{LWE}\rangle$ with gaussian and other amplitudes, 2023. arXiv:2310.00644.

12 Yilei Chen, Qipeng Liu, and Mark Zhandry. Quantum algorithms for variants of average-case lattice problems via filtering. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III*, volume 13277 of *LNCS*, pages 372–401. Springer, 2022. doi:10.1007/978-3-031-07082-2_14.

13 Thomas Debris-Alazard. Code-based cryptography: Lecture notes, arxiv cs.cr 2304.03541, 2023.

14 Thomas Debris-Alazard, Maxime Remaud, and Jean-Pierre Tillich. Quantum reduction of finding short code vectors to the decoding problem. *IEEE Trans. Inform. Theory*, November 2023. in press, see also arXiv:2106.02747 (v2). doi:10.1109/TIT.2023.3327759.

15 Il'ya Dumer. Two decoding algorithms for linear codes. *Probl. Inf. Transm.*, 25(1):17–23, 1989.

16 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998. doi:10.1145/285055.285060.

17 Alex B. Grilo, Iordanis Kerenidis, and Timo Zijlstra. Learning-with-errors problem is easy with quantum samples. *Phys. Rev. A*, 99:032314, March 2019. doi:10.1103/PhysRevA.99.032314.

18 Ghazal Kachigar and Jean-Pierre Tillich. Quantum information set decoding algorithms. In *Post-Quantum Cryptography 2017*, volume 10346 of *LNCS*, pages 69–89, Utrecht, The Netherlands, June 2017. Springer.

19 Alexander May, Alexander Meurer, and Enrico Thomae. Decoding random linear codes in $O(2^{0.054n})$. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology - ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 107–124. Springer, 2011.

20 Alexander May and Ilya Ozerov. On computing nearest neighbors with applications to decoding of binary linear codes. In E. Oswald and M. Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015*, volume 9056 of *LNCS*, pages 203–228. Springer, 2015.

21 Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes, 2012. doi:10.1109/ISIT.2013.6620590.

22 Ashley Montanaro. On the distinguishability of random quantum states. *Communications in Mathematical Physics*, 273, July 2006. doi:10.1007/s00220-007-0221-7.

23 Eugene Prange. The use of information sets in decoding cyclic codes. *IRE Transactions on Information Theory*, 8(5):5–9, 1962. doi:10.1109/TIT.1962.1057777.

24 Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 84–93, 2005. doi:10.1145/1060590.1060603.

25 Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948. doi:10.1002/j.1538-7305.1948.tb01338.x.

26 Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In Mitsuru Matsui, editor, *Advances in Cryptology - ASIACRYPT 2009, 15th International Conference on the Theory and Application of Cryptology and Information Security, Tokyo, Japan, December 6-10, 2009. Proceedings*, volume 5912 of *LNCS*, pages 617–635. Springer, 2009. doi:10.1007/978-3-642-10366-7_36.

27 Jacques Stern. A method for finding codewords of small weight. In G. D. Cohen and J. Wolfmann, editors, *Coding Theory and Applications*, volume 388 of *LNCS*, pages 106–113. Springer, 1988.

28 Takahashi Yamakawa and Mark Zhandry. Verifiable quantum advantage without structure. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 – November 3, 2022*, pages 69–74. IEEE, 2022. doi:10.1109/FOCS54457.2022.00014.

29 Yu Yu, Jiang Zhang, Jian Weng, Chun Guo, and Xiangxue Li. Collision resistant hashing from sub-exponential learning parity with noise. In *ASIACRYPT (2)*, volume 11922 of *Lecture Notes in Computer Science*, pages 3–24. Springer, 2019.

Eigenpath Traversal by Poisson-Distributed Phase Randomisation

Joseph Cunningham
Centre for Quantum Information and Communication (QuIC), Ecole polytechnique de Bruxelles, Université libre de Bruxelles, Belgium

Jérémie Roland
Centre for Quantum Information and Communication (QuIC), Ecole polytechnique de Bruxelles, Université libre de Bruxelles, Belgium

Abstract

We present a framework for quantum computation, similar to Adiabatic Quantum Computation (AQC), that is based on the quantum Zeno effect. By performing randomised dephasing operations at intervals determined by a Poisson process, we are able to track the eigenspace associated to a particular eigenvalue.

We derive a simple differential equation for the fidelity, leading to general theorems bounding the time complexity of a whole class of algorithms. We also use eigenstate filtering to optimise the scaling of the complexity in the error tolerance ϵ.

In many cases the bounds given by our general theorems are optimal, giving a time complexity of $O(1/\Delta_m)$ with Δ_m the minimum of the gap. This allows us to prove optimal results using very general features of problems, minimising the problem-specific insight necessary.

As two applications of our framework, we obtain optimal scaling for the Grover problem (i.e. $O(\sqrt{N})$ where N is the database size) and the Quantum Linear System Problem (i.e. $O(\kappa \log(1/\epsilon))$ where κ is the condition number and ϵ the error tolerance) by direct applications of our theorems.

2012 ACM Subject Classification Theory of computation → Quantum computation theory; Theory of computation → Quantum complexity theory

Keywords and phrases Randomisation method, Non-unitary adiabatic theorems, Grover problem, Quantum linear systems problem

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.7

Funding This work was supported by the Belgian Fonds de la Recherche Scientifique – FNRS under Grants No. R.8015.21 (QOPT) and O.0013.22 (EoS CHEQS).

1 Introduction

It has long been appreciated that the ability to prepare a ground state of a given Hamiltonian is useful for a large number computational tasks. Many NP-hard problems, including various types of partitioning, covering, and satisfiability problems, can be solved by finding the ground state of an Ising system [14]. There are also many applications in the fields of quantum chemistry, where finding the ground state of molecules is a common task, and physics, where knowledge of the ground state helps to understand low-temperature phenomena such as superconductivity and superfluidity.

For computational problems we have the following strategy: (1) find a physical system such that the ground state encodes useful information for solving the problem, (2) prepare the ground state using some physical process and (3) use the information contained in the ground state to solve the problem. This paper is about performing the second step of this strategy.

The most famous way to perform the second step is known as Adiabatic Quantum Computation (AQC) [8]. Suppose H_P is the Hamiltonian whose ground state is of interest. This procedure requires a second Hamiltonian, H_0, with an easily preparable ground state.

Now consider the following interpolated Hamiltonian: $H(s) = (1-s)H_0 + sH_P$ and pick a large time T. We start with the system in the ground state of H_0 and evolve according to the time-dependent Hamiltonian $H(t/T)$, for time $t \in [0, T]$. The adiabatic theorem says that if T is large enough, then the resulting state will be close to the ground state of $H(1) = H_P$. See [11] for results detailing how large T has to be. Clearly we want to take T as small as possible, since a larger T means our computation takes longer.

While AQC is polynomially equivalent to the quantum circuit model [1], it suffers from a few drawbacks. The most significant one being that it requires the system to evolve under a very specific time-dependent Hamiltonian. It is typically very hard to physically implement a system that evolves under exactly this Hamiltonian. Often the complicated time-dependent dynamics are approximated by a sequence of simpler evolutions, which introduces discretisation error. In particular this is necessary when implementing AQC on a conventional quantum computer. Bounding the discretisation error analytically is typically hard to do. In contrast, our method only requires the evolution under a finite number of time-independent Hamiltonians for finite time and thus has no discretisation cost.

There exist alternatives to AQC that are also based on an interpolation $H(s)$ between a Hamiltonian whose ground state is easy to prepare and one whose ground state is difficult to prepare. These approaches use alternate ways to transform the ground state of $H(0)$ into that of $H(1)$, or, more generally some eigenstate of $H(0)$ into the corresponding eigenstate of $H(1)$. They often make use of a variant of the quantum Zeno effect. For instance [5] uses measurement and [4] simulates the quantum Zeno effect by applying Hamiltonian evolutions for random amounts of time in a procedure known as the Randomisation Method (RM).

Our framework builds on the RM of [4] in the following way: instead of performing a fixed sequence of phase randomising steps, we stochastically choose when to perform phase randomisation, based on a Poisson process with rate $\lambda(s)$, see algorithm 1. This has a number of advantages. Firstly it yields a simple differential equation for the state evolution, which fits in the general framework of non-unitary adiabatic theorems of [3], and greatly simplifies the analysis. It allows us to obtain general theorems, see in particular theorems 4 and 5, that in many cases yield optimal results with minimal extra work or problem-specific insight. Also, we only need very minimal technical assumptions on $H(s)$: it only needs to be twice continuously differentiable and we need to know some estimate of the gap between the eigenvalue of interest and the rest of the spectrum. We do not assume precise knowledge of the spectrum or gap. We allow the eigenspace of interest to be degenerate.

Our theorem 4 deals with the case where the rate of the Poisson process λ is taken to be constant. The result we obtain is better than the corresponding result for AQC with a constant-speed linear interpolation. In theorem 5 we describe a variable $\lambda(s)$ that can significantly improve the time complexity, up to $O(1/\Delta_m)$ in the minimum gap Δ_m. Finally theorem 7 improves the dependence of the time complexity on the error tolerance. Typically algorithms based on AQC and the RM have a complexity that scales as $O(1/\epsilon)$ in the error tolerance ϵ. Eigenstate filtering, introduced in [13], can be used to reduce this to $O(\log(1/\epsilon))$. This has been used before to RM-inspired algorithms that use the circuit model, see [13] and [12], but we provide a version native to our cost model.

From theorem 5, we see that the following property is very useful to obtain fast algorithms: $\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = O(\Delta_m^{1-p})$, where Δ is the gap, $\Delta_m = \inf_{s\in[0,1]} \Delta$ and $p > 1$. This property seems to be quite generic, in particular it holds for both the Grover search problem and the Quantum Linear System Problem (QLSP).

In the Grover search problem, [9], the goal is to prepare a specific state in an N-dimensional space with the help of an oracle. It is well-known that this can be done in $O(\sqrt{N})$ queries to the oracle. When AQC was first used to tackle this problem, a complexity of $O(N)$

was obtained [8]. The trick to achieving a complexity of $O(\sqrt{N})$ was to use an adapted schedule [17, 20]. In our framework, the algorithm using constant λ already achieves a scaling of $O\big(\sqrt{N}\log(N)\big)$ by theorem 4, which is significantly better than the corresponding case for AQC. Using a variable $\lambda(s)$, we recover the scaling $O(\sqrt{N})$ by theorem 5. It is interesting to note that, by the generality of our theorems, we actually obtain a whole family of schedules, parametrised by some value $0 < q < 1$, that solve the problem optimally. This is analogous to the range of adiabatic schedules considered in [2] and [7]. The original schedule of [17] actually corresponds to a choice of $q = 1$. It seems like the RM can be considered as the $q = 1$ case of a family of methods, at least in the case of linear interpolation. This falls outside the range of our theorem, but it turns out that $q = 1$ is in fact good enough to give optimality for the Grover problem, which explains why the RM was already known to be able to perform Grover search with optimal complexity $O(\sqrt{N})$, [4].

In general, for other problems, $q = 1$ does not give optimal scaling. The Quantum Linear System Problems (QSLP) is an example of a problem where $q = 1$ does not work, neither in our framework, nor in AQC [2]. In QLSP, [10], the goal is to prepare a quantum state $|x\rangle$ that is proportional to the solution of a system of linear equations $Ax = b$. In [19] the randomisation method was used to construct an algorithm with complexity $O\big(\kappa\log(\kappa)/\epsilon\big)$, where κ is the condition number and ϵ the error tolerance. This is improved to $O\big(\kappa\log(\kappa/\epsilon)\big)$ in [12]. In [7] an algorithm based on a discrete adiabatic theorem was proposed which scales as $O\big(\kappa\log(1/\epsilon)\big)$. We are able to match this in our framework.

There has actually been some discussion recently on the merits of these two approaches to QLSP, i.e. the approach based on the RM of [19] and the approach based on the discrete adiabatic theorem [7]. The approach based on the discrete adiabatic theorem has the better asymptotic scaling, but it turns out that the proven complexity for reasonable values of κ is very large. The paper [12] presents an algorithm that is based on the RM and has a better proven complexity for reasonable values of κ, but is asymptotically suboptimal. Finally [6] uses numerical methods to determine the actual performance of the algorithm based on the discrete adiabatic theorem. They claim that it works much better than the proven bound and in fact better than the algorithm based on the RM. We can contribute to this discussion by noting that our framework gives an algorithm that is based on the RM and has optimal asymptotic scaling. In addition, since the RM seems to correspond to $q = 1$, which we know to be suboptimal, it is likely that the algorithm of [12] can be made asymptotically optimal by changing the scheduling.

1.1 General setup

We assume we have a physical system and a set of (time-independent) Hamiltonians, i.e. self-adjoint operators, such that we can evolve the system under e^{-itH} at a cost of t for any Hamiltonian H in this set.[1] We call the Hamiltonians in this set admissible. Which Hamiltonians are admissible will depend on the device or setup, but typically they will be bounded in norm.

This is not the cost model used by references [7] and [12], which use a query complexity rather than a time complexity. We discuss a translation of our results to this model using optimal Hamiltonian simulation in appendix B. The asymptotic complexities are mostly unaffected, but there are different constants involved.

For a given instance of a problem, we assume that we have a continuous, twice differentiable

[1] We set $\hbar = 1$.

path of admissible Hamiltonians $H(s)$, where $s \in [0,1]$. We also assume that we can prepare the ground state of $H(0)$.

We are interested in the asymptotic scaling of the time complexity, as measured by the total length of time we apply unitaries of the form e^{-itH}. We produce theorems that give bounds on the complexity in terms of the spectral gap and the derivatives $\|H'\|$ and $\|H''\|$.

Our main tool will be the randomised application of unitaries of the form e^{-itH}. Since we are using classical randomness, it will be useful to use the density matrix formalism. Using this formalism, we can derive differential equations for these new procedures that share essential features with the Liouville–von Neumann equation in the adiabatic limit. This allows us to use many of the same mathematical tricks to study these procedures and we can derive "generalised" adiabatic theorems in the sense of [3].

1.1.1 Cost and error model

It is clear what the cost of one run of the algorithm is: it is just the total time spent evolving the system under some Hamiltonian. In order to state the time complexity we have the additional problem that the running time of the algorithm is not deterministic. That is, even for a fixed input, multiple runs of the algorithm will take different amounts of time. Our time complexity uses the expected run time of each input. Thus we say our algorithm has time complexity T if, for all relevant inputs I, the expected time taken by the algorithm with input I is less than T. In other words, we may consider this a worst-case expected-time complexity.

In order to guarantee that the algorithm does not take too long, we could abort if the chosen amount of time was too long. This would yield an additional error, which can be bounded by Markov's inequality.

Our algorithms are also not guaranteed to give the correct answer, rather we aim to produce the target state with at least a certain target fidelity.

1.1.2 Technical assumptions on the spectrum

We assume the existence of the following objects: a number $\Delta_m > 0$ and functions $\omega_0 : [0,1] \to \mathbb{R}$ and $\Delta : [0,1] \to [0,1]$ such that

- ω_0 continuous;
- $\omega_0(s)$ is an eigenvalue of $H(s)$ for all $s \in [0,1]$;
- $\Delta(s) \geq \Delta_m$ for all $s \in [0,1]$;
- the intersection of $[\omega_0(s) - \Delta(s), \omega_0(s) + \Delta(s)]$ with the spectrum of $H(s)$ is exactly $\{\omega_0(s)\}$.

Let $P(s)$ be the projector on the eigenspace associated to the eigenvalue $\omega(s)$. We also set $Q(s) = \mathrm{id} - P(s)$.

In order to perform our algorithm, we assume knowledge of Δ, which bounds the gap. We do not assume more detailed knowledge of the gap, ω_0, or any other part of the spectrum.

2 Poisson-distributed phase randomisation

Our algorithms are built using a finite number of steps, where at each step a Zeno-like dephasing operation is performed. This dephasing operation is given by the following proposition:

▶ **Proposition 1** (Phase randomisation [4]). *Let H be a Hamiltonian and ω_0, Δ, P and Q as above. Assume we can simulate e^{-itH} for any positive or negative time t at a cost of $|t|$. Then we can construct a stochastic variable τ such that for all states ρ,*

$$\langle e^{-i\tau H}\rho e^{i\tau H}\rangle = P\rho P + Q\langle e^{-i\tau H}\rho e^{i\tau H}\rangle Q, \tag{1}$$

with cost $\langle|\tau|\rangle = t_0/\Delta$, where $t_0 = 2.32132$.

The angled brackets mean taking the average over τ. The result is originally from [4]. The value for t_0 was obtained in theorem 2 of [12].

The algorithm is now simple to state:

Algorithm 1 Poisson-distributed phase randomisation.

```
1 Pick a Poisson process $N : [0,1] \times (\Omega, \mathcal{A}, P) \to \mathbb{N}$ with rate $\lambda(s)$;
2 At each jump point $s$ of the Poisson process, pick an instance $t$ of the random
    variable $T$ as defined in proposition 1 and evolve the system under the Hamiltonian
    evolution $e^{-itH(s)}$;
```

The density matrix describing the system is a random variable that satisfies the stochastic differential equation $\mathrm{d}\rho = \left(e^{-i\tau(s)H(s)}\rho e^{i\tau(s)H(s)} - \rho\right)\mathrm{d}N$. Averaging over realisations, we get

$$\mathrm{d}\langle\rho\rangle = \left(P\langle\rho\rangle P + Q\langle e^{-i\tau H}\rho e^{i\tau H}\rangle Q - \langle\rho\rangle\right)\lambda\,\mathrm{d}s. \tag{2}$$

Note that this should properly be thought of as a "marginalised" density matrix, rather than an "average" density matrix. This is entirely analoguous to the situation for classical probability distributions, where integrating out a variable gives the marginal distribution. In this case we are marginalising over the choice of Poisson process N. In the rest of the paper, we will use ρ to refer to the marginalised distribution $\langle\rho\rangle$. This corresponds to the density matrix you would observe if you were not told which Poisson process N was chosen.

The total time taken by one run of the algorithm is a random variable T satisfying

$$\mathrm{d}T = \tau\,\mathrm{d}N. \tag{3}$$

In order to find the time complexity, we take the average. This gives $\mathrm{d}T = t_0\Delta^{-1}\lambda\,\mathrm{d}s$, so $T = \int_0^1 \frac{t_0\lambda}{\Delta}\,\mathrm{d}s$.

2.1 Analysis

▶ **Lemma 2.** *Under the assumptions in 1.1.2, the algorithm 1 with rate $\lambda(s)$ produces a state with an infidelity that is bounded by*

$$\epsilon \leq \left\|\lambda(0)^{-1}P'(0)\right\| + \left\|\lambda(1)^{-1}P'(1)\right\| + \int_0^1 \left(\left\|\frac{P''}{\lambda}\right\| + \left|\left(\frac{1}{\lambda}\right)'\right|\|P'\|\right)\mathrm{d}s. \tag{4}$$

Proof. The infidelity is given by $\epsilon = 1 - \operatorname{Tr}\left(P(1)\rho(1)\right) = \operatorname{Tr}\left(P(0)\rho(0)\right) - \operatorname{Tr}\left(P(1)\rho(1)\right) = \left|\operatorname{Tr}(P\rho)\big|_0^1\right|$, so it makes sense to track how the fidelity $\operatorname{Tr}\left(P(s)\rho(s)\right)$ changes in time.

We construct a differential equation for $\operatorname{Tr}(P\rho)$ by taking the derivative with respect to s, $\operatorname{Tr}(P\rho)' = \operatorname{Tr}(P'\rho) + \operatorname{Tr}(P\rho')$. This can be simplified using the fact that $PP'P = 0$ and $QP'Q = 0$.[2] Indeed, we have

$$\operatorname{Tr}(P\rho') = \lambda\operatorname{Tr}\left(P(P\rho P + Q\langle e^{-i\tau H}\rho e^{i\tau H}\rangle Q - \rho)\right) = \operatorname{Tr}(P\rho P) - \operatorname{Tr}(P\rho) = 0 \tag{5}$$

[2] We have $P' = (PP)' = P'P + PP'$, so $PP'P = 2PP'P$ and $QP'Q = 0$.

and

$$\begin{aligned}
\operatorname{Tr}(P'\rho) &= \operatorname{Tr}\left(P'(P\rho P + Q\langle e^{-i\tau H}\rho e^{i\tau H}\rangle Q - \lambda^{-1}\rho')\right) && (6)\\
&= \operatorname{Tr}(PP'P\rho) + \operatorname{Tr}\left((QP'Q)\langle e^{-i\tau H}\rho e^{i\tau H}\rangle\right) - \operatorname{Tr}(\lambda^{-1}P'\rho') && (7)\\
&= -\operatorname{Tr}(\lambda^{-1}P'\rho'), && (8)
\end{aligned}$$

so $\operatorname{Tr}(P\rho)' = -\operatorname{Tr}(\lambda^{-1}P'\rho')$. Integrating gives

$$\begin{aligned}
\operatorname{Tr}(P\rho)\big|_0^1 &= -\int_0^1 \operatorname{Tr}(\lambda^{-1}P'\rho')\,\mathrm{d}s && (9)\\
&= -\lambda^{-1}\operatorname{Tr}(P'\rho)\big|_0^1 + \int_0^1 \left(\operatorname{Tr}\left(\frac{P''}{\lambda}\rho\right) + \operatorname{Tr}\left((\tfrac{1}{\lambda})'P'\rho\right)\right)\mathrm{d}s, && (10)
\end{aligned}$$

which we can bound by

$$\begin{aligned}
\epsilon = \left|\operatorname{Tr}(P\rho)\big|_0^1\right| &\le \left|\operatorname{Tr}(\lambda^{-1}P'\rho)\big|_0^1\right| + \int_0^1 \left(\operatorname{Tr}\left(\left|\frac{P''}{\lambda}\rho\right|\right) + \left|\left(\frac{1}{\lambda}\right)'\right|\operatorname{Tr}\left(\left|P'\rho\right|\right)\right)\mathrm{d}s && (11)\\
&\le \left(\left\|\lambda(0)^{-1}P'(0)\right\| + \left\|\lambda(1)^{-1}P'(1)\right\|\right)\operatorname{Tr}(\rho)\\
&\quad + \int_0^1 \left(\left\|\frac{P''}{\lambda}\right\| + \left|\left(\frac{1}{\lambda}\right)'\right|\|P'\|\right)\operatorname{Tr}(\rho)\,\mathrm{d}s && (12)\\
&\le \left\|\lambda(0)^{-1}P'(0)\right\| + \left\|\lambda(1)^{-1}P'(1)\right\|\\
&\quad + \int_0^1 \left(\left\|\frac{P''}{\lambda}\right\| + \left|\left(\frac{1}{\lambda}\right)'\right|\|P'\|\right)\mathrm{d}s. && (13)
\end{aligned}$$

◀

The next step is to bound $\|P'\|$ and $\|P''\|$ by more useful quantities. We make use of the following lemma:

▶ **Lemma 3.** *Under the assumptions stated in 1.1.2, we have*

1. $\|P'\| \le 2\frac{\|H'\|}{\Delta}$;
2. $\|P''\| \le 8\frac{\|H'\|^2}{\Delta^2} + 2\frac{\|H''\|}{\Delta}$.

This is fairly standard. See for example [16]. A proof is provided in appendix A. We are now ready to use lemma 2 in two distinct contexts, leading to theorems 4 and 5.

2.1.1 Constant λ

We first derive a theorem under the assumption that λ is constant. In this case we obtain the following result:

▶ **Theorem 4.** *Under the assumptions in 1.1.2, the algorithm 1 produces the target state with fidelity of at least $1-\epsilon$ if λ is constant and*

$$\epsilon^{-1}2\Big(\frac{\|H'(0)\|}{\Delta(0)} + \frac{\|H'(1)\|}{\Delta(1)} + \int_0^1 4\frac{\|H'\|^2}{\Delta^2} + \frac{\|H''\|}{\Delta}\,\mathrm{d}s\Big) \le \lambda. \tag{14}$$

In this case the time complexity of the procedure is given by $T = \lambda t_0 \int_0^1 \frac{1}{\Delta}\,\mathrm{d}s$.

Proof. Let ϵ_0 be the actual error of the algorithm. We need $\epsilon_0 \leq \epsilon$. We can use lemma 3 to rewrite the inequality in lemma 2 as

$$\epsilon_0 \leq \lambda^{-1}\big(\|P'(0)\| + \|P'(1)\|\big) + \lambda^{-1}\int_0^1 \|P''\|\,\mathrm{d}s \tag{15}$$

$$\leq \lambda^{-1}\Big(2\frac{\|H'(0)\|}{\Delta(0)} + 2\frac{\|H'(1)\|}{\Delta(1)} + \int_0^1 8\frac{\|H'\|^2}{\Delta^2} + 2\frac{\|H''\|}{\Delta}\,\mathrm{d}s\Big). \tag{16}$$

Set $B := 2\Big(\frac{\|H'(0)\|}{\Delta(0)} + \frac{\|H'(1)\|}{\Delta(1)} + \int_0^1 4\frac{\|H'\|^2}{\Delta^2} + \frac{\|H''\|}{\Delta}\,\mathrm{d}s\Big)$. Then we have

$$\epsilon_0 \leq \lambda^{-1}B \leq \epsilon B^{-1}B = \epsilon, \tag{17}$$

so the algorithm works. The time complexity is simply given by $T = \int_0^1 \frac{t_0\lambda}{\Delta}\,\mathrm{d}s = \lambda t_0 \int_0^1 \frac{1}{\Delta}\,\mathrm{d}s$. ◀

This result can be compared to theorem 15 in the circuit model.

2.1.2 Scaling λ with the gap

We know from [17] and [20] that the performance of AQC can be improved with an adapted schedule, taking more time when the gap is small. Similarly we expect it to be possible to improve the performance of our procedure by varying λ. Indeed this is the case.

▶ **Theorem 5.** *Under the assumptions in 1.1.2, we additionally assume that there exists $0 \leq q \leq 1$ and B_1, B_2 such that $\int_0^1 \frac{1}{\Delta^{1+q}}\,\mathrm{d}s \leq B_1\Delta_m^{-q}$ and $\int_0^1 \frac{1}{\Delta^{2-q}}\,\mathrm{d}s \leq B_2\Delta_m^{q-1}$ for all instances of the problem. Then algorithm 1 produces the target state with a fidelity of at least $1-\epsilon$ if*

$$\lambda = \epsilon^{-1}\frac{C}{\Delta^q\Delta_m^{1-q}}, \tag{18}$$

where $C := 2\sup_{s\in[0,1]}\Big(2\|H'(s)\| + 4\|H'(s)\|^2 B_2 + \|H''(s)\| + q|\Delta'(s)|\,\|H'(s)\|\,B_2\Big)$.

In this case the time complexity of the procedure is given by

$$T \leq \epsilon^{-1}\frac{t_0 B_1 C}{\Delta_m}.$$

▶ **Corollary 6.** *If $\int_0^1 \frac{1}{\Delta^p}\,\mathrm{d}s = O(\Delta_m^{1-p})$ holds for all $p > 1$, $|\Delta'| = O(1)$, $\|H'\| = O(1)$ and $\|H''\| = O(1)$, then algorithm 1 with the rate defined in theorem 5 produces the target state with fidelity $1-\epsilon$ and a time complexity of $O(\Delta_m^{-1})$ for all $0 < q < 1$.*

Proof of Theorem 5. Let ϵ_0 be the actual error of the algorithm, we need $\epsilon_0 \leq \epsilon$. In this case the inequality in lemma 2 becomes

$$\epsilon_0 \leq \epsilon C^{-1}\Delta_m^{1-q}\big(\Delta(0)^q\left\|P'(0)\right\| + \Delta(1)^q\left\|P'(1)\right\|\big) + \epsilon C^{-1}\int_0^1 \Delta^q\Delta_m^{1-q}\left\|P''\right\| + \left|\Big(\Delta^q\Delta_m^{1-q}\Big)'\right|\left\|P'\right\|\mathrm{d}s. \tag{19}$$

We bound the terms separately, using lemma 3. For the first, we have

$$\Delta_m^{1-q}\Delta^q\left\|P'\right\| \leq 2\Delta_m^{1-q}\Delta^q\frac{\|H'\|}{\Delta} = 2\Delta_m^{1-q}\frac{\|H'\|}{\Delta^{1-q}} \leq 2\Delta_m^{1-q}\frac{\|H'\|}{\Delta_m^{1-q}} = 2\left\|H'\right\| \leq 2\sup_{s\in[0,1]}\left\|H'\right\| \tag{20}$$

at both $s = 0$ and $s = 1$, so we bound the sum by $4 \sup_{s\in[0,1]} \|H'\|$.

The second term splits into two, since we bound $\|P''\|$ by $8\frac{\|H'\|^2}{\Delta^2} + 2\frac{\|H''\|}{\Delta}$. For the first part we have

$$8\int_0^1 \Delta^q \Delta_m^{1-q} \frac{\|H'\|^2}{\Delta^2}\,\mathrm{d}s \le 8 \sup_{s\in[0,1]} \|H'(s)\|^2 \Delta_m^{1-q} \int_0^1 \frac{1}{\Delta^{2-q}}\,\mathrm{d}s \tag{21}$$

$$\le 8 \sup_{s\in[0,1]} \|H'(s)\|^2 B_2 \Delta_m^{1-q} \Delta_m^{q-1} = 8 \sup_{s\in[0,1]} \|H'(s)\|^2 B_2. \tag{22}$$

The second part gives

$$2\int_0^1 \Delta^q \Delta_m^{1-q} \frac{\|H''\|}{\Delta}\,\mathrm{d}s \le 2 \sup_{s\in[0,1]} \|H''(s)\| \Delta_m^{1-q} \int_0^1 \frac{1}{\Delta^{1-q}}\,\mathrm{d}s \tag{23}$$

$$\le 2 \sup_{s\in[0,1]} \|H''(s)\| \Delta_m^{1-q} \Delta_m^{q-1} = 2 \sup_{s\in[0,1]} \|H''(s)\|. \tag{24}$$

Finally, for the third term,

$$\int_0^1 \left|\left(\Delta^q \Delta_m^{1-q}\right)'\right| \|P'\|\,\mathrm{d}s = \int_0^1 q\Delta^{q-1}\Delta_m^{1-q}|\Delta'|\,\|P'\|\,\mathrm{d}s \tag{25}$$

$$\le 2q\Delta_m^{1-q}\Big(\sup_{s\in[0,1]} |\Delta'(s)|\,\|H'(s)\|\Big)\int_0^1 \frac{\Delta^{q-1}}{\Delta}\,\mathrm{d}s \tag{26}$$

$$= 2q\Delta_m^{1-q}\Big(\sup_{s\in[0,1]} |\Delta'(s)|\,\|H'(s)\|\Big)\int_0^1 \frac{1}{\Delta^{2-q}}\,\mathrm{d}s \tag{27}$$

$$\le 2q\Big(\sup_{s\in[0,1]} |\Delta'(s)|\,\|H'(s)\|\Big)\Delta_m^{1-q} B_2 \Delta_m^{q-1} \tag{28}$$

$$= 2qB_2\Big(\sup_{s\in[0,1]} |\Delta'(s)|\,\|H'(s)\|\Big). \tag{29}$$

Plugging everything back into equation (19), gives

$$\epsilon_0 \le \epsilon C^{-1} \sup_{s\in[0,1]} \Big(4\|H'(s)\| + 8\|H'(s)\|^2 B_2 + 2\|H''(s)\| + 2q|\Delta'(s)|\,\|H'(s)\| B_2\Big) \tag{30}$$

$$= \epsilon C^{-1} C = \epsilon, \tag{31}$$

so the procedure works. We can then calculate the time complexity

$$T = \int_0^1 \frac{t_0\lambda}{\Delta}\,\mathrm{d}s = \epsilon^{-1}t_0\int_0^1 \frac{C}{\Delta^q\Delta_m^{1-q}\Delta}\,\mathrm{d}s = \epsilon^{-1}t_0 C\Delta_m^{q-1}\int_0^1 \frac{1}{\Delta^{q+1}}\,\mathrm{d}s$$

$$\le \epsilon^{-1}t_0 C\Delta_m^{q-1}B_1\Delta_m^{-q} = \epsilon^{-1}t_0 C B_1 \Delta_m^{-1}. \tag{32}$$

◀

These results can be compared to theorem 16 and corollary 17 in the circuit model.

3 Improving the scaling in the error with eigenstate filtering

The use of eigenstate filtering was introduced in [13] to improve scaling in the error tolerance for algorithms based on adiabatic principles and the quantum Zeno effect, in particular with application to QLSP.

A similar technique was used in [7] to achieve optimal scaling, but using Linear Combinations of Unitaries (LCU) instead of Quantum Signal Processing (QSP). We adapt the technique of [7] to the present situation.

▶ **Theorem 7.** *Let H be a Hamiltonian with $\|H\| \leq 1$ and 0 in the spectrum of H, $\sigma(H)$. Suppose*

- *$\Delta \geq 0$ is such that $[-\Delta, \Delta] \cap \sigma(H) = \{0\}$;*
- *P is the orthogonal projector on the eigenspace associated to the eigenvalue 0, we set $Q := \mathbb{1} - P$;*
- *ρ is a density matrix of the form $P\rho_0 P + Q\rho_1 Q$ with $\mathrm{Tr}(P\rho_0) > 1/2$, that we can prepare at cost T_0;*
- *$\epsilon > 0$.*

Further, suppose

- *we can adjoin two ancilla qubits to ρ;*
- *we can can measure and reprepare the ancilla qubits;*
- *we can evolve the system under $H \otimes R$ and $\mathbb{1} \otimes R$ for time t for all Hermitian operators R on $\mathbb{C}^{2\times 2}$ with $\|R\| \leq 1$ at a cost of t.*

Then we can prepare a state ρ_2 such that $\mathrm{Tr}(P\rho_2) \geq 1-\epsilon$ at a cost of $T = O(T_0 + \Delta^{-1}\log(1/\epsilon))$.

The idea of the procedure is relatively simple. With these assumptions, we can apply controlled versions of the unitary e^{-itH}, i.e. $e^{itH\otimes\Pi}$ for some projector Π on $\mathbb{C}^{2\times 2}$. This means that we can apply linear combinations of $e^{itH\otimes\Pi}$ using the technique of linear combinations of unitaries, see lemmas 8 and 9. In particular we can apply a polynomial that has a large peak at 0 and is very small everywhere else. We use this to filter out the part of the state that we do not want.

▶ **Lemma 8** (LCU with arbitrarily large ancilla register)**.** *Let $f(x) = \sum_{k=-n}^{n} a_k x^k$ be a rational polynomial with complex coefficients such that $\sum_{k=-n}^{n} |a_k|^2 = 1$. Let H be a Hamiltonian and ρ the state of the system. Assume we have access to an ancilla register with orthonormal basis $\{|k\rangle \mid k \in \mathbb{Z}\}$. Then, at a cost of $O(nt)$, we can do an operation which either*

- *succeeds and applies $\sum_{k=-n}^{n} |a_k|^2 e^{-itkH}$ to the system,*
- *or fails, with a probability of $1 - \mathrm{Tr}\left(\left(\sum_{k=-n}^{n} |a_k|^2 e^{-itkH}\right)\rho\left(\sum_{k=-n}^{n} |a_k|^2 e^{itkH}\right)\right)$. We can see when this has happened thanks to the measured contents of the ancilla register.*

Proof. The procedure is as follows: we first prepare the ancilla in the state $|f\rangle := \sum_{k=-n}^{n} a_k|k\rangle$, then apply $\sum_{k=-n}^{n} kH \otimes |k\rangle\langle k|$ for time t and finally measure the state $|f\rangle$. If we measure any other state than $|f\rangle$, the procedure fails.

The result then follows from the following identity:

$$\sum_{k,l=-n}^{n} \Big(\mathbb{1} \otimes a_k \langle k|\Big) e^{-it\sum_{m=-n}^{n} mH\otimes|m\rangle\langle m|} \Big(\mathbb{1} \otimes \overline{a_l}|l\rangle\Big) = \sum_{m} |a_m|^2 e^{-itmH}. \tag{33}$$

Defining

$$\Pi_m^0 = \mathbb{1} - \sum_{k=0}^{m} |k\rangle\langle k| \qquad \text{and} \qquad \Pi_m^1 = \mathbb{1} - \sum_{k=-m}^{0} |k\rangle\langle k|, \tag{34}$$

we can write $e^{-it\sum_{m=-n}^{n} mH\otimes|m\rangle\langle m|} = \prod_{m=0}^{n-1} e^{-itH\otimes\Pi_m^0} e^{itH\otimes\Pi_m^1}$, which we can clearly apply at a cost of $2nt$.

The cost of ancilla preparation depends on the admissible operations on the ancilla register, but in a worst-case scenario, each a_k needs to be set separately[3] which means that the cost is $O(n)$. The total cost is then still $O(nt)$. ◀

[3] This is the case for the procedure used in lemma 9.

▶ **Lemma 9** (LCU with two ancilla qubits). *We can achieve the results of 8 only using two ancilla qubits at a time.*

The construction is identical to the one in [7].

Proof of theorem 7. Let $Q \coloneqq \mathbb{1} - P$ and write $Q = \sum_j Q_j$, where each Q_j is an eigenprojector of H associated to the eigenvalue ω_j. Now we observe

$$\Big(\sum_{k=-n}^{n} |a_k|^2 e^{-ikH}\Big) Q_j = \Big(\sum_{k=-n}^{n} |a_k|^2 e^{-ik\omega_j}\Big) Q_j = A(\omega_j) Q_j, \tag{35}$$

where $A(\omega)$ is the Fourier transform of the sequence $|a_k|^2$. Thus

$$\Big(\sum_{k=-n}^{n} |a_k|^2 e^{-ikH}\Big) Q\rho Q \Big(\sum_{k=-n}^{n} |a_k|^2 e^{ikH}\Big) = \sum_{j,l} \Big(\sum_{k=-n}^{n} |a_k|^2 e^{-ikH}\Big) Q_j \rho Q_l \Big(\sum_{k=-n}^{n} |a_k|^2 e^{ikH}\Big) \tag{36}$$

$$= \sum_{j,l} A(\omega_j) A(-\omega_l) Q_j \rho Q_l. \tag{37}$$

Taking the trace gives $\mathrm{Tr}\Big(\sum_{j,l} A(\omega_j)A(-\omega_l)Q_j\rho Q_l\Big) \leq \max_{\omega\notin[-\Delta,\Delta]} A(\omega)^2\,\mathrm{Tr}(Q\rho Q) \leq \max_{\omega\notin[-\Delta,\Delta]} A(\omega)^2$. The goal then becomes to find a sequence and its Fourier transform such that $A(\omega_0) = 1$, $\max_{\omega\notin[-\Delta,\Delta]} A(\omega)^2 \leq \epsilon$ and whose window n is as small as possible. The answer to this optimisation problem is well-known and is given by the Dolph-Chebyshev window [15]. In this case we need a window of[4]

$$n = \frac{\cosh^{-1}(1/\sqrt{\epsilon})}{\cosh^{-1}\big(\sec(\Delta)\big)} \leq \frac{1}{2\Delta}\log\Big(\frac{4}{\epsilon}\Big). \tag{38}$$

By lemma 8, we can implement this at a cost of $O(n)$. Note that this procedure terminates succesfully with a probability of at least $\mathrm{Tr}(P\rho_0)$ (which is bounded below) and we can check to see whether the procedure failed. If it failed, we repeat. On average we need to repeat fewer than $\mathrm{Tr}(P\rho_0)^{-1}$ times, which is $O(1)$. ◀

4 Applications

4.1 Grover search

For the Grover problem, we have an N-dimensional vector space we want to find an element of an M-dimensional subspace $\mathcal{M}$. In order to help us, we assume we have access to an oracle Hamiltonian $H_1 = \mathbb{1} - P_{\mathcal{M}}$, where $P_{\mathcal{M}}$ is the orthogonal projector on $\mathcal{M}$. In other words, we assume H_1 is admissible. We also assume $H_0 = \mathbb{1} - |u\rangle\langle u|$ is admissible, where $|u\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N}|i\rangle$ is the uniform superposition. The aim is now to use the interpolation $H(s) = (1-s)H_0 + sH_1$ to prepare as state in $\mathcal{M}$. For more details see [8] and [18].

We see that $H(s)$ has four eigenvalues:

$$\lambda_{1,2} = \frac{1}{2}\left(1 \pm \sqrt{1 - 4(1-\frac{M}{N})s(1-s)}\right) \qquad \text{with multiplicity } 1 \tag{39}$$

$$\lambda_3 = 1 - s \qquad \text{with multiplicity } M-1 \tag{40}$$

$$\lambda_4 = 1 \qquad \text{with multiplicity } N-M-1. \tag{41}$$

[4] We note that we improve the scaling by a factor of two compared to [7]. This is because we are able to start from a state where $P\rho Q = 0 = Q\rho P$.

The eigenvectors corresponding to λ_3 are the eigenvectors in $\mathcal{M}$ with zero overlap with $|u\rangle$. The eigenvectors corresponding to λ_4 are the eigenvectors in $\mathcal{M}^\perp$ with zero overlap with $|u\rangle$. Since the initial state has zero overlap with any of these vectors and they are eigenvectors of each $H(s)$, none of them are prepared by the procedure and everything happens in the two-dimensional space spanned by the eigenvectors associated to λ_1 and λ_2.

We have explicitly computed the gap, so we can use this as the bound Δ:

$$\Delta(s) = \sqrt{1 - 4(1-\frac{M}{N})s(1-s)}. \tag{42}$$

We can set $\Delta_m = \min_{s\in[0,1]} \Delta(s) = \sqrt{M/N}$. In order to give bounds on the time-complexity, we use the following result:

▶ **Lemma 10.** *For all $p > 1$ and Δ given by (42), we have*

$$\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = O\big(\sqrt{N/M}^{p-1}\big) = O\big(\Delta_m^{1-p}\big), \tag{43}$$

and, for $p = 1$,

$$\int_0^1 \frac{1}{\Delta(s)}\,\mathrm{d}s = O\big(\log(N/M)\big). \tag{44}$$

We provide a proof in appendix C.1. For constant λ, we apply theorem 4 and use the lemma 10 to get a time complexity $O\big(\sqrt{N/M}\log(N/M)\big)$.

We are able to take the q in corollary 6 to be anywhere in the range $0 < q < 1$, since for any such q both $1 + q$ and $2 - q$ are strictly greater than 1. This is related to the range of schedules described in [2]. The time complexity of the algorithm for any such q is $O(\sqrt{N/M})$, since it is easy to check that to other conditions hold: $\|H'\| = \|H_1 - H_0\|$, $\|H''\| = 0$ and

$$\begin{aligned} |\Delta'| &= \left|\frac{4(1-\frac{M}{N})(\frac{1}{2}-s)}{\Delta}\right| \qquad (45)\\ &\le \frac{2\sqrt{4(1-\frac{M}{N})(\frac{1}{2}-s)^2}}{\Delta} \qquad (46)\\ &\le \frac{2\sqrt{\frac{M}{N} + 4(1-\frac{M}{N})(\frac{1}{2}-s)^2}}{\Delta} = 2\frac{\Delta}{\Delta} = 2. \qquad (47)\end{aligned}$$

4.2 Solving linear systems of equations

The Quantum Linear Systems Problem (QLSP) was introduced in [10]. Suppose A is an invertible $N \times N$ matrix $b \in \mathbb{C}^N$ a vector. The goal is to prepare the quantum state $\frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}$. We express the time complexity of our algorithm in terms of the condition number $\kappa = \|A\|\,\|A^{-1}\|$.

We may restrict ourselves to Hermitian matrices because we can use the following trick from [10]: If A is not Hermitian, we consider the matrix $\begin{pmatrix} 0 & A \\ A^* & 0 \end{pmatrix}$, which has the same condition number, and solve the equation $\begin{pmatrix} 0 & A \\ A^* & 0 \end{pmatrix}|y\rangle = \begin{pmatrix} |b\rangle \\ 0 \end{pmatrix}$.

First we rescale the matrix A to $\frac{A}{\|A\|}$. We do this because typically admissible matrices need to be uniformly bounded. This has the effect of shifting the lowest singular value from $\frac{1}{\|A^{-1}\|}$ to $\frac{1}{\|A\|\|A^{-1}\|} = \kappa^{-1}$. Now we consider a path of Hamiltonians that was introduced in

[19]. Define $A(s) \coloneqq (1-s)\sigma_z \otimes \mathbb{1} + s\sigma_x \otimes A$, $Q_{b,+} \coloneqq \mathbb{1} - \big(|+\rangle|b\rangle\big)\big(\langle+|\langle b|\big)$ and $\sigma_\pm \coloneqq \frac{1}{2}\big(\sigma_x \pm i\sigma_y\big)$. Set $H(s) = \sigma_+ \otimes \big(A(s)Q_{b,+}\big) + \sigma_- \otimes \big(Q_{b,+}A(s)\big)$. This can be written as a linear interpolation $H(s) = (1-s)H_0 + sH_1$, where

$$H_0 \coloneqq \sigma_+ \otimes \big((\sigma_z \otimes \mathbb{1})Q_{b,+}\big) + \sigma_- \otimes \big(Q_{b,+}(\sigma_z \otimes \mathbb{1})\big) \tag{48}$$

$$H_1 \coloneqq \sigma_+ \otimes \big((\sigma_x \otimes A)Q_{b,+}\big) + \sigma_- \otimes \big(Q_{b,+}(\sigma_x \otimes A)\big). \tag{49}$$

Following the analysis of [19], we see that $H(s)$ has 0 as an eigenvalue for all $s \in [0,1]$. The corresponding eigenspace is spanned by $\{|0\rangle \otimes |x(s)\rangle, |1\rangle \otimes |+\rangle|b\rangle\}$, where $|x(s)\rangle \coloneqq \frac{A(s)^{-1}|b\rangle}{\|A(s)^{-1}|b\rangle\|}$. Since $H(s)$ does not allow transition between these states, we are sure to not prepare $|1\rangle \otimes |+\rangle|b\rangle$, so long as we start with $|0\rangle \otimes |x(0)\rangle$.

In [19] it was also shown that the eigenvalue zero is separated from the rest of the spectrum by a gap that is at least

$$\Delta(s) = \sqrt{(1-s)^2 + \left(\frac{s}{\kappa}\right)^2}. \tag{50}$$

If κ is large enough, then we can take $\Delta_m \coloneqq \frac{1}{2\kappa} \leq \sqrt{\frac{1}{\kappa^2+1}} = \min_{s\in[0,1]} \Delta(s)$.

In order to give bounds on the time-complexity, we use the following result:

▶ **Lemma 11.** *For all $p > 1$, we have*

$$\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = O\big(\kappa^{p-1}\big) = O\big(\Delta_m^{1-p}\big), \tag{51}$$

and, for $p = 1$,

$$\int_0^1 \frac{1}{\Delta(s)}\,\mathrm{d}s = O\big(\log(\kappa)\big). \tag{52}$$

We provide a proof in appendix C.2.

For constant λ, we apply theorem 4 and use the lemma 11 to get a time complexity $O\big(\kappa\log(\kappa)\big)$. This is also the complexity that was obtained in [19].

As before, we have a full order reduction for $p > 1$ and thus we are able to take the q in corollary 6 to be anywhere in the range $0 < q < 1$, since for any such q both $1+q$ and $2-q$ are strictly greater than 1. If $q = 0$ or $q = 1$, the complexity gains a factor of $\log(\kappa)$. This exactly mirrors the situation in [2] and is the reason why the algorithms for QLSP based on the RM have an extra factor of $\log(\kappa)$ in the asymptotic complexity, see [19] and [12].

We can apply 6 since $\|H'\| = \|H_1 - H_0\|$, $\|H''\| = 0$ and

$$|\Delta'| = \left|\frac{s-1+s/\kappa^2}{\Delta}\right| \tag{53}$$

$$= \frac{\sqrt{(s-1+s/\kappa^2)^2}}{\Delta} \tag{54}$$

$$= \frac{\sqrt{(1+1/\kappa^2)^2 s^2 - (1+1/\kappa^2)2s + 1}}{\Delta} \tag{55}$$

$$\leq \frac{\sqrt{(1+1/\kappa^2)^2 s^2 - (1+1/\kappa^2)2s + (1+1/\kappa^2)}}{\Delta} \tag{56}$$

$$= \sqrt{1+1/\kappa^2}\,\frac{\Delta}{\Delta} = \sqrt{1+1/\kappa^2} = O(1). \tag{57}$$

This yields a time complexity of $O(\kappa)$ for fixed error tolerance. The scaling on both condition number and error tolerance is $O(\epsilon^{-1}\kappa)$. By a straightforward application of theorem 7 at $s = 1$, we get a scaling of $O\big(\log(\epsilon^{-1})\kappa\big)$. This is possible, since we know the eigenvalue of interest is 0.

This result is optimal and matches the complexity reported in [7], where it was achieved using a very different method.

References

1 Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. Adiabatic quantum computation is equivalent to standard quantum computation. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 42–51, 2004. doi:10.1109/FOCS.2004.8.

2 Dong An and Lin Lin. Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm. *ACM Transactions on Quantum Computing*, 3(2):1–28, March 2022. doi:10.1145/3498331.

3 Joseph E. Avron, Martin Fraas, Gian M. Graf, and Philip Grech. Adiabatic theorems for generators of contracting evolutions. *Communications in Mathematical Physics*, 314(1):163–191, May 2012. doi:10.1007/s00220-012-1504-1.

4 Sergio Boixo, Emanuel Knill, and Rolando Somma. Eigenpath traversal by phase randomization. *Quantum Information and Computation*, 9(9&10):833–855, September 2009. doi:10.26421/QIC9.9-10-7.

5 Andrew M. Childs, Enrico Deotto, Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Andrew J. Landahl. Quantum search by measurement. *Physical Review A*, 66(3), September 2002. doi:10.1103/physreva.66.032314.

6 Pedro C. S. Costa, Dong An, Ryan Babbush, and Dominic Berry. The discrete adiabatic quantum linear system solver has lower constant factors than the randomized adiabatic solver, 2023. arXiv:2312.07690.

7 Pedro C.S. Costa, Dong An, Yuval R. Sanders, Yuan Su, Ryan Babbush, and Dominic W. Berry. Optimal scaling quantum linear-systems solver via discrete adiabatic theorem. *PRX Quantum*, 3:040303, October 2022. doi:10.1103/PRXQuantum.3.040303.

8 Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution, 2000. arXiv:quant-ph/0001106.

9 Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 212–219, New York, NY, USA, 1996. Association for Computing Machinery. doi:10.1145/237814.237866.

10 Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15), October 2009. doi:10.1103/physrevlett.103.150502.

11 Sabine Jansen, Mary-Beth Ruskai, and Ruedi Seiler. Bounds for the adiabatic approximation with applications to quantum computation. *Journal of Mathematical Physics*, 48(10), October 2007. doi:10.1063/1.2798382.

12 David Jennings, Matteo Lostaglio, Sam Pallister, Andrew T Sornborger, and Yiğit Subaşı. Efficient quantum linear solver algorithm with detailed running costs, 2023. arXiv:2305.11352.

13 Lin Lin and Yu Tong. Optimal polynomial based quantum eigenstate filtering with application to solving quantum linear systems. *Quantum*, 4:361, November 2020. doi:10.22331/q-2020-11-11-361.

14 Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2, 2014. doi:10.3389/fphy.2014.00005.

15 Peter Lynch. The dolph–chebyshev window: A simple optimal filter. *Monthly Weather Review*, 125(4):655–660, 1997. doi:10.1175/1520-0493(1997)125<0655:TDCWAS>2.0.CO;2.

16 Ben W. Reichardt. The quantum adiabatic optimization algorithm and local minima. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '04, pages 502–510, New York, NY, USA, 2004. Association for Computing Machinery. doi:10.1145/1007352.1007428.

17 Jérémie Roland and Nicolas J. Cerf. Quantum search by local adiabatic evolution. *Physical Review A*, 65(4), March 2002. doi:10.1103/physreva.65.042308.

18 Jérémie Roland and Nicolas J. Cerf. Quantum-circuit model of hamiltonian search algorithms. *Physical Review A*, 68(6), December 2003. doi:10.1103/physreva.68.062311.

19 Yiğit Subaşı, Rolando D. Somma, and Davide Orsucci. Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing. *Physical Review Letters*, 122(6), February 2019. doi:10.1103/physrevlett.122.060504.

20 Wim van Dam, Michele Mosca, and Umesh Vazirani. How powerful is adiabatic quantum computation? In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*. IEEE, 2001. doi:10.1109/sfcs.2001.959902.

A Bounds on derivatives of projectors

We provide a proof of lemma 3

▶ **Lemma.** *Under the assumptions stated in 1.1.2, we have*

1. $\|P'\| \leq 2\frac{\|H'\|}{\Delta}$;
2. $\|P''\| \leq 8\frac{\|H'\|^2}{\Delta^2} + 2\frac{\|H''\|}{\Delta}$.

Proof. Let Γ be a circle in the complex plane, centred at the ground energy with radius $\Delta/2$. Then we have the Riesz form of the projector

$$P = \frac{1}{2\pi i}\oint_\Gamma R_H(z)\,\mathrm{d}z,$$

where $R_H(z) = \big(z\,\mathrm{id} - H\big)^{-1}$ is the resolvent of H at z. Then $R_H(z)' = R_H(z)H'R_H(z)$ (the derivative is with respect to s, not z). As H is a normal operator, the norm $\|R_H(z)\|$ is equal to the inverse of the distance from z to the spectrum $\sigma(H)$. On the circle Γ this is equal to $(\Delta/2)^{-1}$ everywhere. We can then approximate

$$\begin{aligned}\|P'\| &= \left\|\frac{1}{2\pi i}\oint_\Gamma R_H(z)'\,\mathrm{d}z\right\| \\ &\leq \frac{1}{2\pi}\oint_\Gamma \|R_H(z)'\|\mathrm{d}z \\ &\leq \frac{1}{2\pi}\oint_\Gamma \|R_H(z)\|\cdot\|H'\|\cdot\|R_H(z)\|\,\mathrm{d}z \\ &= \frac{1}{2\pi}\left(\frac{2}{\Delta}\right)^2\|H'\|\oint_\Gamma \mathrm{d}z \\ &= \frac{1}{2\pi}\left(\frac{2}{\Delta}\right)^2 2\pi\frac{\Delta}{2}\|H'\| \\ &= 2\frac{\|H'\|}{\Delta}.\end{aligned}$$

Similarly, we can write

$$P'' = \frac{1}{2\pi i}\oint_\Gamma 2R_H(z)H'R_H(z)H'R_H(z) + R_H(z)H''R_H(z)\,\mathrm{d}z.$$

Estimating this in the same way as before yields

$$\|P''\| \leq 8\frac{\|H'\|^2}{\Delta^2} + 2\frac{\|H''\|}{\Delta}.$$ ◀

B Comparison with the circuit model

So far we have assumed access to a device that can evolve a system under a given Hamiltonian in real time, i.e. it takes time t to apply e^{-itH}. This is the typical setting of AQC and is also the setting of [4] and [19].

Many papers use a slightly different setup. In [7] and [12] the setting is a standard quantum computer which is given access to block encodings of $H(s)$ for all $s \in [0,1]$. In this case the complexity is given by the number of times such a block-encoded Hamiltonian is used. In other words, the complexity is a query complexity rather than a time complexity.

Given access to only a block encoding of a Hamiltonian H, it is generally not possible to simulate e^{-itH} exactly. Instead, we can use proposition 12, which is taken from [12].

▶ **Proposition 12** (Theorem 4 from [12]). *Given access to an $(\alpha, m, 0)$–block-encoding U_H of a Hermitian operator H with $\|H\| \leq 1$, we can realise a $(1, m+2, \delta)$–block-encoding of e^{-itH} for $t \in \mathbb{R}$ with*

$$3\left\lceil \frac{e}{2}\alpha|t| + \log\left(\frac{2c}{\delta}\right)\right\rceil$$

calls to U_H, U_H^* with $c = 4(\sqrt{2\pi}e^{\frac{1}{13}})^{-1} \approx 1.47762$.

In this proposition δ gives the error of the block encoding, i.e. if U is the block encoding, then $\left\|U - e^{-itH}\right\| \leq \delta$.

This motivates replacing the algorithm 1 by the algorithm 2, which now depends on both a rate $\lambda(s)$ and an allowable simulation error $\delta(s)$.

Algorithm 2 Poisson-distributed phase randomisation with imperfect time evolution.

```
Pick a Poisson process $N : [0,1] \times (\Omega, \mathcal{A}, P) \to \mathbb{N}$ with rate $\lambda(s)$;
At each jump point $s$ of the Poisson process, pick an instance $t$ of the random variable
  $T$ as defined in proposition 1 and use proposition 12 to simulate the evolution under
  the time-independent Hamiltonian $H(s)$ for a time $t$ with error at most $\delta(s)$;
```

In this case the number of queries is bounded by a quantity Q is a random variable with stochastic differential equation $\mathrm{d}Q = 3\Big(\frac{e}{2}\alpha|\tau| + \log\left(\frac{2c}{\delta}\right) + 1\Big)\,\mathrm{d}N$. Taking the average yields $Q = 3\int_0^1 \left(\frac{e\alpha t_0}{2\Delta} + \log\left(\frac{2c}{\delta}\right) + 1\right)\lambda\,\mathrm{d}s$.

To analyse algorithm 2, we prove lemma 13, which is analogous to lemma 2.

▶ **Lemma 13.** *Given the assumptions in 1.1.2 and that for all $s \in [0,1]$ and $t \in [0, \infty[$, we can apply an operation $A(s,t)$ such that $\left\|e^{-itH(s)} - A(s,t)\right\| \leq \delta(s)$, the algorithm 1 with rate $\lambda(s)$ has an error that is bounded by*

$$\epsilon \leq \int_0^1 (2\delta+\delta^2)\left(2\frac{\|H'\|}{\Delta}+\lambda(s)\right)\mathrm{d}s + \left\|\lambda(0)^{-1}P'(0)\right\| + \left\|\lambda(1)^{-1}P'(1)\right\| + \int_0^1 \left(\left\|\frac{P''}{\lambda}\right\| + \left|\left(\frac{1}{\lambda}\right)'\right| \left\|P'\right\|\right)\mathrm{d}s. \tag{58}$$

Proof. The differential equation (2) is then

$$\rho' = \lambda\Big(P\rho P + Q\langle e^{-i\tau H}\rho e^{i\tau H}\rangle Q - \rho\Big) + \lambda\Big(A(s,t)\rho A(s,t)^* - e^{-i\tau H}\rho e^{i\tau H}\Big). \tag{59}$$

We set $E := A(s,t)\rho A(s,t)^* - e^{-i\tau H}\rho e^{i\tau H}$. Then equations (5) and (8) become $\operatorname{Tr}(P\rho') = \lambda\operatorname{Tr}(E)$ and $\operatorname{Tr}(P'\rho) = \operatorname{Tr}(P'E) - \operatorname{Tr}(\lambda^{-1}P'\rho')$, so

$$\operatorname{Tr}(P\rho)' = \operatorname{Tr}(P\rho') + \operatorname{Tr}(P'\rho) = \lambda\operatorname{Tr}(E) + \operatorname{Tr}(P'E) - \operatorname{Tr}(\lambda^{-1}P'\rho'). \tag{60}$$

The integral of $-\operatorname{Tr}(\lambda^{-1}P'\rho')$ was bounded in the proof of lemma 2.

Using lemma 14, we bound $\operatorname{Tr}(|E|) \leq 2\delta + \delta^2$. Together with the bound $\|P'\| \leq 2\frac{\|H'\|}{\Delta}$, this yields the result. ◀

In this proof we have made use of the following lemma:

▶ **Lemma 14** (Lemma 9 from [12]). *Suppose A, B are operators such that $\|A - B\| \leq \delta$. Then, for each density operators ρ, we have*

$$\operatorname{Tr}(|A\rho A^* - B\rho B^*|) \leq 2\delta \|A\| + \delta^2. \tag{61}$$

▶ **Theorem 15.** *Under the assumptions in 1.1.2, the algorithm 2 produces the target state with fidelity of at least $1 - \epsilon$ if λ is constant, $\delta = \frac{4\epsilon}{27\lambda}$ and*

$$\epsilon^{-1}4\Big(\frac{\|H'(0)\|}{\Delta(0)} + \frac{\|H'(1)\|}{\Delta(1)} + \int_0^1 4\frac{\|H'\|^2}{\Delta^2} + \frac{\|H''\|}{\Delta}\,\mathrm{d}s\Big) \leq \lambda. \tag{62}$$

Using the Hamiltonian simulation of proposition 12, this gives a query complexity of

$$Q = \lambda\Big(e\alpha t_0 \frac{3}{2}\int_0^1 \frac{1}{\Delta}\,\mathrm{d}s + 3\log\big(\frac{27c}{2\epsilon}\big) + \log(\lambda) + 1\Big).$$

Proof. Let ϵ_0 be the actual error of the algorithm. We need $\epsilon_0 \leq \epsilon$. As in the proof of 4, rewrite the inequality in lemma 13 as

$$\epsilon_0 \leq (2\delta+\delta^2)\Big(\int_0^1 2\frac{\|H'\|}{\Delta}\,\mathrm{d}s+\lambda\Big)+\lambda^{-1}\Big(2\frac{\|H'(0)\|}{\Delta(0)}+2\frac{\|H'(1)\|}{\Delta(1)}+\int_0^1 8\frac{\|H'\|^2}{\Delta^2}+2\frac{\|H''\|}{\Delta}\,\mathrm{d}s\Big). \tag{63}$$

As in the proof of 4, the second term is bounded by $\epsilon/2$. Since all the terms of equation (62) are positive, we have $\int_0^1 2\frac{\|H'\|}{\Delta}\,\mathrm{d}s \leq \frac{\epsilon\lambda}{8}$. Then we bound

$$(2\delta + \delta^2)\Big(\int_0^1 2\frac{\|H'\|}{\Delta}\,\mathrm{d}s + \lambda\Big) \leq 3\delta\big(\frac{\epsilon\lambda}{8} + \lambda\big) \tag{64}$$

$$\leq \frac{27}{8}\delta\lambda(\epsilon + 1) \leq \frac{27}{8}\delta\lambda \leq \frac{\epsilon}{2}. \tag{65}$$

Finally we consider the query complexity and calculate

$$Q = 3\int_0^1 \Big(\frac{e\alpha t_0}{2\Delta} + \log\big(\frac{2c}{\delta}\big) + 1\Big)\lambda\,\mathrm{d}s \tag{66}$$

$$\leq \lambda\Big(e\alpha t_0\frac{3}{2}\int_0^1 \frac{1}{\Delta}\,\mathrm{d}s + 3\log\big(\frac{27c}{2\epsilon}\big) + \log(\lambda) + 1\Big). \tag{67}$$

◀

▶ **Theorem 16.** *Under the assumptions in 1.1.2, we additionally assume that there exists $0 \leq q \leq 1$ and B_1, B_2 such that $\int_0^1 \frac{1}{\Delta^{1+q}}\,\mathrm{d}s \leq B_1\Delta_m^{-q}$ and $\int_0^1 \frac{1}{\Delta^{2-q}}\,\mathrm{d}s \leq B_2\Delta_m^{q-1}$ for all instances of the problem. Then algorithm 2 produces the target state with a fidelity of at least $1 - \epsilon$ if*

$$\lambda = \epsilon^{-1}\frac{2C}{\Delta^q\Delta_m^{1-q}} \tag{68}$$

$$\delta = \frac{2\epsilon}{15\lambda} \tag{69}$$

where $C := 2\sup_{s\in[0,1]}\Big(2\|H'(s)\| + 4\|H'(s)\|^2 B_2 + \|H''(s)\| + q|\Delta'(s)|\ \|H'(s)\| B_2\Big)$.

If, in addition, there exists a constant B_3 *such that* $\int_0^1 \frac{1}{\Delta^{2q}}\,\mathrm{d}s \leq B_3\Delta_m^{1-2q}$ *and the Hamiltonian simulation of 12 is used, this gives a query complexity of*

$$Q \leq \frac{1}{\epsilon\Delta_m}\Big(12C\log(\epsilon^{-1}) + 3e\alpha t_0 CB_1 + 6\log(15c)C + 12C^2B_3\Big).$$

Proof. As before, we need to bound the inequality in lemma 13. Everything except the first term has already been bounded in the proof of theorem 5 to be less than $\epsilon/2$. (Notice that we are taking λ to be twice the rate specified in theorem 5.)

We now need to show that the first term in the inequality in lemma 13 can be bounded by $\epsilon/2$. Indeed, we calculate

$$\int_0^1 (2\delta+\delta^2)\Big(2\frac{\|H'\|}{\Delta} + \lambda(s)\Big)\,\mathrm{d}s \leq \int_0^1 3\delta\Big(\frac{C}{2\Delta} + \lambda\Big)\,\mathrm{d}s \tag{70}$$

$$= \int_0^1 \frac{2\epsilon}{5\lambda}\Big(\frac{C}{2\Delta} + \lambda\Big)\,\mathrm{d}s \tag{71}$$

$$= \frac{2\epsilon}{5}\Big(1 + \int_0^1 \epsilon\frac{\Delta_m^{q-1}}{4\Delta^{q-1}}\,\mathrm{d}s\Big) \tag{72}$$

$$\leq \frac{2\epsilon}{5}\Big(1 + \frac{\epsilon}{4}\Big) \leq \frac{\epsilon}{2}. \tag{73}$$

Finally we consider the query complexity and calculate, using the fact that $\log(x)+1 \leq x$ for all positive x,

$$Q = 3\int_0^1 \Big(\frac{e\alpha t_0}{2\Delta} + \log\big(\frac{2c}{\delta}\big) + 1\Big)\lambda\,\mathrm{d}s \tag{74}$$

$$= 3\int_0^1 \Big(\frac{e\alpha t_0}{2\Delta} + \log\big(\frac{15c}{\epsilon}\big) + \log(\epsilon\lambda) + 1\Big)\lambda\,\mathrm{d}s \tag{75}$$

$$\leq 3\int_0^1 \Big(\frac{e\alpha t_0}{2\Delta}\lambda + \log\big(\frac{15c}{\epsilon^2}\big)\lambda + \epsilon\lambda^2\Big)\,\mathrm{d}s. \tag{76}$$

We bound each term separately. First

$$3\int_0^1 \frac{e\alpha t_0}{2\Delta}\lambda\,\mathrm{d}s = \frac{3e\alpha t_0 C}{\epsilon\Delta_m^{1-q}}\int_0^1 \frac{1}{\Delta^{q+1}}\lambda\,\mathrm{d}s \tag{77}$$

$$\leq \frac{3e\alpha t_0 C}{\epsilon\Delta_m^{1-q}}B_1\Delta_m^{-q} = \frac{3e\alpha t_0 CB_1}{\epsilon\Delta_m}. \tag{78}$$

Next

$$3\int_0^1 \log\big(\frac{15c}{\epsilon^2}\big)\lambda\,\mathrm{d}s = 3\log\big(\frac{15c}{\epsilon^2}\big)\epsilon^{-1}\int_0^1 \frac{2C}{\Delta^q\Delta_m^{1-q}}\,\mathrm{d}s \tag{79}$$

$$\leq \log\big(\frac{15c}{\epsilon^2}\big)\epsilon^{-1}\frac{6C}{\Delta_m}. \tag{80}$$

Finally

$$3\int_0^1 \epsilon\lambda^2\,\mathrm{d}s = \frac{12C^2}{\epsilon\Delta_m^{2-2q}}\int_0^1 \frac{1}{\Delta^{2q}}\,\mathrm{d}s \tag{81}$$

$$\leq \frac{12C^2B_3}{\epsilon\Delta_m^{2-2q}}\Delta_m^{1-2q} = \frac{12C^2B_3}{\epsilon\Delta_m}. \tag{82}$$

Putting everything together yields the query complexity. ◀

▶ **Corollary 17.** *If $\int_0^1 \frac{1}{\Delta^p}\,\mathrm{d}s = O(\Delta_m^{1-p})$ holds for all $p > 1$, $|\Delta'| = O(1)$, $\|H'\| = O(1)$ and $\|H''\| = O(1)$, then algorithm 2 with the Hamiltonian simulation of 12 and the parameters of 16 for some $1/2 < q < 1$, produces a state with fidelity larger than $1 - \epsilon$ using a number of queries that scales as $O\big(\Delta_m^{-1}\epsilon^{-1}\log(\epsilon^{-1})\big)$.*

The asymptotic scaling in the error ϵ is slightly worse here, since there is an extra logarithmic factor, but this is not an issue if we want to apply eigenstate filtering. With eigenstate filtering the scaling in the error is still $O(\log(1/\epsilon))$.

C Gap properties

C.1 The gap in the Grover problem

For the Grover problem we have the following gap:

$$\Delta(s) = \sqrt{1 - 4(1 - \frac{M}{N})s(1-s)}. \tag{83}$$

We can set $\Delta_m = \min_{s\in[0,1]} \Delta(s) = \sqrt{M/N}$. We provide a proof of lemma 10.

▶ **Lemma.** *For all $p > 1$ and Δ given by* (83)*, we have*

$$\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = O\big(\sqrt{N/M}^{p-1}\big) = O\big(\Delta_m^{1-p}\big), \tag{84}$$

and, for $p = 1$,

$$\int_0^1 \frac{1}{\Delta(s)}\,\mathrm{d}s = O\big(\log(N/M)\big). \tag{85}$$

Proof. We note that $\Delta(s)$ is symmetric about $s = 1/2$. It is also strictly decreasing on $[0, 1/2]$, going from 1 to a minimum of $\sqrt{M/N}$. So we can write

$$\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = 2\int_0^{1/2} \frac{1}{\Delta(s)^p}\,\mathrm{d}s \tag{86}$$

$$= 2\Big(\int_0^{1/2-\sqrt{M/N}} \frac{1}{\Delta(s)^p}\,\mathrm{d}s + \int_{1/2-\sqrt{M/N}}^{1/2} \frac{1}{\Delta(s)^p}\,\mathrm{d}s\Big). \tag{87}$$

Since Δ has a minimum of $\sqrt{M/N}$, we can bound the second integral by

$$\int_{1/2-\sqrt{M/N}}^{1/2} \frac{1}{\Delta(s)^p}\,\mathrm{d}s \leq \sqrt{\frac{M}{N}}\Big(\frac{1}{\min_{s\in[0,1]}\Delta(s)}\Big)^p = \frac{\sqrt{M/N}}{\sqrt{M/N}^p} = \sqrt{N/M}^{p-1}.$$

For the first integral, we write

$$\int_0^{1/2-\sqrt{M/N}} \frac{1}{\Delta(s)^p}\,\mathrm{d}s = \int_1^{\Delta\left(1/2-\sqrt{M/N}\right)} \frac{1}{\Delta^p}\frac{\mathrm{d}s}{\mathrm{d}\Delta}\,\mathrm{d}\Delta \tag{88}$$

$$= \int_{\Delta\left(1/2-\sqrt{M/N}\right)}^{1} \frac{1}{\Delta^p}\Big(-\frac{\mathrm{d}s}{\mathrm{d}\Delta}\Big)\,\mathrm{d}\Delta. \tag{89}$$

We can invert (42) to obtain $s = \frac{1}{2} - \frac{1}{2}\sqrt{1 - \frac{1-\Delta^2}{1-N/M}}$. Then we have

$$-\tfrac{\mathrm{d}s}{\mathrm{d}\Delta} = \frac{\Delta}{2\sqrt{(1-M/N)(\Delta^2 - M/N)}}. \tag{90}$$

We now calculate

$$\Delta\Big(\frac{1}{2}-\sqrt{\frac{M}{N}}\Big)=\sqrt{\frac{M}{N}}\sqrt{5-4\frac{M}{N}}\geq 2\sqrt{\frac{M}{N}},$$

assuming $M/N \leq 1/4$. So

$$\int_0^{1/2-\sqrt{M/N}}\frac{1}{\Delta^p}\,\mathrm{d}s \leq \int_{2\sqrt{\frac{M}{N}}}^1 \frac{1}{\Delta^p}\Big(-\frac{\mathrm{d}s}{\mathrm{d}\Delta}\Big)\,\mathrm{d}\Delta \tag{91}$$

$$= \int_{2\sqrt{\frac{M}{N}}}^1 \frac{1}{\Delta^p}\frac{\Delta}{2\sqrt{(1-M/N)(\Delta^2-M/N)}}\,\mathrm{d}\Delta \tag{92}$$

$$\leq \int_{2\sqrt{\frac{M}{N}}}^1 \frac{1}{\Delta^p}\frac{\Delta}{2\sqrt{(1-M/N)(\Delta^2-\Delta^2/4)}}\,\mathrm{d}\Delta \tag{93}$$

$$= \frac{1}{\sqrt{3(1-M/N)}}\int_{2\sqrt{\frac{M}{N}}}^1 \frac{1}{\Delta^p}\,\mathrm{d}\Delta. \tag{94}$$

Now $\frac{1}{\sqrt{3(1-M/N)}}$ is $O(1)$ and $\int_{2\sqrt{\frac{M}{N}}}^1 \frac{1}{\Delta^p}\,\mathrm{d}\Delta = \Big[\frac{1}{(p-1)\Delta^{p-1}}\Big]_{2\sqrt{M/N}}^1$ is $O\big(\sqrt{N/M}^{p-1}\big)$, if $p>1$. If $p=1$, then it is $O\big(\log\sqrt{N/M}\big)$. ◀

C.2 The gap in QLSP

For the quantum linear system problem we have the following bound on the gap:

$$\Delta(s)=\sqrt{(1-s)^2+\Big(\frac{s}{\kappa}\Big)^2}. \tag{95}$$

If κ is large enough, then we can take $\Delta_m := \frac{1}{2\kappa} \leq \sqrt{\frac{1}{\kappa^2+1}} = \min_{s\in[0,1]}\Delta(s)$. We provide a proof of lemma 11.

▶ **Lemma.** *For all $p>1$, we have*

$$\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = O\big(\kappa^{p-1}\big) = O\big(\Delta_m^{1-p}\big), \tag{96}$$

and, for $p=1$,

$$\int_0^1 \frac{1}{\Delta(s)}\,\mathrm{d}s = O\big(\log(\kappa)\big). \tag{97}$$

Proof. We note that $\Delta(s)$ is strictly decreasing on $\Big[0, 1-\frac{1}{\kappa^2+1}\Big]$, going from 1 to a minimum of $\sqrt{\frac{1}{\kappa^2+1}}$. So we can write

$$\int_0^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s = \int_0^{1-\frac{1}{\kappa^2+1}} \frac{1}{\Delta(s)^p}\,\mathrm{d}s + \int_{1-\frac{1}{\kappa^2+1}}^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s.$$

Since Δ has a minimum of $\sqrt{\frac{1}{\kappa^2+1}}$, we can bound the second integral by

$$\int_{1-\frac{1}{\kappa^2+1}}^1 \frac{1}{\Delta(s)^p}\,\mathrm{d}s \leq \frac{1}{\kappa^2+1}\Big(\frac{1}{\min_{s\in[0,1]}\Delta(s)}\Big)^p = \frac{1}{\kappa^2+1}\big(\kappa^2+1\big)^{p/2} = \big(\kappa^2+1\big)^{p/2-1}.$$

For the first integral, we write

$$\int_0^{1-\frac{1}{\kappa^2+1}} \frac{1}{\Delta^p}\,\mathrm{d}s = \int_1^{\Delta\left(1-\frac{1}{\kappa^2+1}\right)} \frac{1}{\Delta^p}\frac{\mathrm{d}s}{\mathrm{d}\Delta}\,\mathrm{d}\Delta \tag{98}$$

$$= \int_{\Delta\left(1-\frac{1}{\kappa^2+1}\right)}^1 \frac{1}{\Delta^p}\Big(-\frac{\mathrm{d}s}{\mathrm{d}\Delta}\Big)\,\mathrm{d}\Delta \tag{99}$$

$$= \int_{\sqrt{\frac{1}{\kappa^2+1}}}^1 \frac{1}{\Delta^p}\Big(-\frac{\mathrm{d}s}{\mathrm{d}\Delta}\Big)\,\mathrm{d}\Delta. \tag{100}$$

We can invert (95) on $\Big[0, 1-\frac{1}{\kappa^2+1}\Big]$ to obtain $s = \frac{\kappa^2}{\kappa^2+1}(1-\Delta)$. Then we have

$$-\tfrac{\mathrm{d}s}{\mathrm{d}\Delta} = \frac{\kappa^2}{\kappa^2+1}, \tag{101}$$

so

$$\int_0^{1-\frac{1}{\kappa^2+1}} \frac{1}{\Delta^p}\,\mathrm{d}s = \int_{\sqrt{\frac{1}{\kappa^2+1}}}^1 \frac{1}{\Delta^p}\frac{\kappa^2}{\kappa^2+1}\,\mathrm{d}\Delta \tag{102}$$

$$= \frac{\kappa^2}{\kappa^2+1}\Big(\frac{1}{(p-1)\Delta^{p-1}}\Big)\Big|_{\Delta=1}^{\Delta=\sqrt{\frac{1}{\kappa^2+1}}} \tag{103}$$

$$= O(\kappa^{p-1}). \tag{104}$$

If $p = 1$, then the integral is $O\big(\log(\kappa)\big)$. ◀

(Quantum) Complexity of Testing Signed Graph Clusterability

Kuo-Chin Chen
Hon Hai Research Institute, Taipei, Taiwan

Simon Apers
Université de Paris, CNRS, IRIF, France

Min-Hsiu Hsieh
Hon Hai Research Institute, Taipei, Taiwan

Abstract

This study examines clusterability testing for a signed graph in the bounded-degree model. Our contributions are two-fold. First, we provide a quantum algorithm with query complexity $\tilde{O}(N^{1/3})$ for testing clusterability, which yields a polynomial speedup over the best classical clusterability tester known [1]. Second, we prove an $\tilde{\Omega}(\sqrt{N})$ classical query lower bound for testing clusterability, which nearly matches the upper bound from [1]. This settles the classical query complexity of clusterability testing, and it shows that our quantum algorithm has an advantage over any classical algorithm.

2012 ACM Subject Classification Theory of computation → Proof complexity; Theory of computation → Graph algorithms analysis

Keywords and phrases Quantum Algorithm, classical Query lower Bound, Graph Property testing

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.8

Related Version *Previous Version*: `https://arxiv.org/abs/2311.10480`

1 Introduction

Property testing [24, 39] deals with the setting where we wish to distinguish between objects, e.g., functions [2, 8, 32] or graphs [5, 7, 6, 9, 12, 3], that satisfy a predetermined property and those that are far from satisfying this property. For certain properties, this relaxed setting allows algorithms to query only a small part of (sometimes huge) data sets. Indeed, the goal in property testing is to design so-called *property testers* to solve a property testing problem within sublinear time complexity. Property testing has been studied in many settings, such as computational learning theory [25, 13, 38, 27, 20, 22], quantum information theory [35, 16, 17, 15, 36, 10], coding theory [23, 41, 29, 31, 34, 37, 21], and so on. This witnesses the significant attention that property testing has drawn from the academic community.

An interesting setting is that of graph property testing. In the *dense graph model*, it was shown that a constant number of queries are needed to test a wide range of graph partition properties [25], including k-colorability, ρ-clique, and ρ-cut for any fixed $k \geq 2$ and $\rho > 0$. For comparison, in the *bounded-degree model* [26] similar graph properties such as bipartiteness and expansion testing require sublinear $\tilde{\Theta}(\sqrt{N})$ classical queries. Moreover, some graph properties even have a (trivial) $\Omega(N)$ query complexity, as Ref. [14] showed for 3-colorability in the bounded-degree model. While there have been numerous studies on testing graph properties, there has been little work on testing the properties of *signed* graphs.

A signed graph is a graph where each edge is assigned a positive or a negative label. They can be applied to model a variety of problems including correlation clustering problems [13, 19], modeling the ground state energy of Ising models [30], and social network problems [28, 33, 40]. Signed graphs have different properties than unsigned graphs. One of these is

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 8; pp. 8:1–8:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

the important property of clusterability, which was introduced by Davis [18] to describe the correlation clustering problem. We call a signed graph clusterable if it can be decomposed into several components such that (1) the edges in each component are all positive, and (2) the edges connecting the vertices belonging to different components are all negative. This property is equivalent to not having a "bad cycle", which is a cycle with exactly one negative edge [18]. An algorithm for testing clusterability in the bounded-degree model with only $\tilde{O}(\sqrt{N})$ queries was proposed in [1]. The optimality of this clusterability tester was left as an open question. Here, we prove that any classical algorithm requires at least $\tilde{\Omega}(\sqrt{N})$ queries to test clusterability, showing that the tester from [1] is nearly-optimal.

As a natural extension of past studies, we are interested in whether quantum computing can provide any advantages in testing clusterability for signed graphs. To the best of our knowledge, we are not aware of any previous work on the quantum advantage for testing the properties of signed graphs. However, in work by Ambainis, Childs and Liu [11], a quantum speedup for testing bipartiteness and expansion of bounded-degree graphs was shown. We adopt these techniques to obtain a quantum algorithm for testing clusterability in signed graphs. More specifically, we combine their quantum approach with the classical property testing techniques provided by Adriaens and Apers [1] to obtain a quantum algorithm for testing the clusterability of bounded-degree graphs in time $\tilde{O}(N^{1/3})$. This outperforms the $\tilde{O}(\sqrt{N})$ query complexity of the classical tester in [1] (which is optimal by our lower bound). We leave optimality of the quantum algorithm for testing clusterability as an open question. Indeed, settling the quantum query complexity of property testing in the bounded-degree graph model has been a long open question, and even for the well-studied problem of bipartiteness testing no matching lower bound is known [11].

1.1 Overview of Main Results

Here we formally state our main results (precise definitions are deferred to Section 2). First, we prove a lower bound on the classical query complexity of clusterability testing for a signed graph.

▶ **Theorem 2** (Restated). *Any classical clusterability tester with error parameter $\epsilon = 0.01$ must make at least $\sqrt{N}/10$ queries.*

Up to polylogarithmic factors this matches the upper bound from [1], thus proving that their clusterability tester is optimal in the classical computing regime. However, taking inspiration from this classical clusterability tester, we reduce the clusterability testing problem to a collision finding problem which can be solved faster by quantum computing. As a result, we propose a quantum clusterability tester with a query complexity $\tilde{O}(N^{1/3})$.

▶ **Theorem 6** (Restated). *We propose a quantum clusterability tester with query complexity $\tilde{O}(N^{1/3})$.*

This improves over the classical lower bound, implying a quantum advantage over classical algorithms for testing clusterability.

1.2 Technical contributions

A sketch of the proof of our two results is given in this section. The first result is the classical query lower bound for testing clusterability. While the bound follows the blueprint of the lower bound for bipartiteness testing by Goldreich and Ron [26], we have to deal with a number of additional complications in the signed graph setting.

The main idea of the lower bound is to show that, with less than $\sqrt{N}/10$ queries, we cannot distinguish two families of graphs with degree d: one family $\mathcal{G}_1^N$ that is ϵ-far from clusterable, and another family $\mathcal{G}_2^N$ that is clusterable. The design of these two families of graphs must take into account two constraints. The first constraint is that the graphs in $\mathcal{G}_2^N$ cannot contain a bad cycle, while those in $\mathcal{G}_1^N$ must have at least one bad cycle, even if we remove ϵNd edges of the graph. This ensures that $\mathcal{G}_2^N$ is clusterable, while $\mathcal{G}_1^N$ is far from clusterable. The second constraint relates to the fact that both graph should be locally indistinguishable. This requires for instance that vertices in each graph in both families are incident to the same number of positive and negative edges. If in addition we can ensure that each cycle in these graphs contains many edges with a constant probability, then we can use this to show that no algorithm can distinguish the graphs in these two families with $o(\sqrt{N})$ queries. Indeed, we show that these two families of graphs are indistinguishable with less than $\sqrt{N}/10$ queries as follows. First, we propose two random processes P_1 and P_2, one generates a uniformly random graph in $\mathcal{G}_1^N$, and the other generates a uniformly random graph in $\mathcal{G}_2^N$. Specifically, P_α for $\alpha \in \{1,2\}$ takes a query given from an algorithm as input and returns a vertex while "on-the-fly" or "lazily" constructing a graph from $\mathcal{G}_\alpha^N$. In other words, P_α simulates how an algorithm interacts with a graph sampled uniformly in $\mathcal{G}_\alpha^N$. We observe that these random processes are statistically identical if the answer to each query is not found in the past answers or queries, which is equivalent to not finding a cycle when exploring a graph. Second, we demonstrate that the probability of these random processes being statistically identical is greater than $1/4$ within $\sqrt{N}/10$ queries. In other words, no classical algorithm can distinguish between these two families with a probability exceeding $3/4$ within $\sqrt{N}/10$ queries to the input graph.

Our second result is a quantum algorithm for clusterability testing with a better query complexity. To this end, we reduce the main procedure in the algorithm proposed by Adriaens and Apers [1] to a collision finding algorithm. This collision finding problem can then be solved using the quantum collision finding algorithm, similar to [11]. The main idea is that if we implement several random walks on the positive edges of a graph that is far from clusterable, then there exists a negative edge between the vertices belonging to distinct random walks with a constant probability. We define finding such a negative edge between random walks as finding a collision, a process that can be solved by using a quantum collision finding algorithm. This yields a quantum speedup for clusterability testing.

2 Preliminaries

This section contains two parts. Section 2.1 defines some of the basic terminology associated with the graphs used in this paper. In Section 2.2, we introduce the graph clusterability testing problem.

2.1 Terminology

A graph $G = (V, E)$ is a pair of sets. The elements in $V = [N]$ are vertices, and the elements in E, denoted by edges, are paired vertices. The vertices $v \in V$ and $u \in V$ of an edge $(v, u) \in E$ are the endpoints of (v, u), and (v, u) is incident to u and v. The vertices u and v are adjacent if there exist an edge $(v, u) \in E$. The number of edges incident with v, denoted by $d(v)$, is the degree of a vertex, and the maximum degree among the vertices in G is the degree of the graph $G(V, E)$.

Given a graph $G = (V, E)$, a walk is a sequence of edges $((v_1, v_2), (v_2, v_3), \cdots, (v_{J-1}, v_J))$ where $(v_j, v_{j+1}) \in E$ for all $1 \leq j \leq J-1$ and $v_j \in V$ for all $1 \leq j \leq J$. This walk can also be denoted as a sequence of vertices $(v_1, v_2, \ldots, v_J)$. A trail is a walk in which all edges are distinct. A cycle is a non-empty trail in which only the first and last vertices are equal. A Hamiltonian cycle is a cycle of a graph in which every vertex of the graph is visited exactly once.

A signed graph $G = (V, E, \Sigma)$ consists of the vertex set V, the edge set $E \subseteq V \times V$, and a mapping $\Sigma : E \to \{+, -\}$ that indicates the sign of each edge. We say that a signed graph $G = (V, E, \Sigma)$ is clusterable if we can partition vertices into components such that (i) every edge that connects two vertices in the same components is positive, and (ii) every edge that connects two vertices in different components is negative.

2.2 Clusterability testing for signed graphs

We can easily modify the usual graph query model to signed graphs. Given a signed graph G with maximum degree d, the bounded-degree graph model is defined as follows. A query is a tuple (v, i) where $v \in [N]$ is a vertex in the graph and $i \in [d]$. The oracle answers this query with (i) the i^{th} neighbor of the vertex v if the degree of v is at least i (otherwise it returns an error symbol), and (ii) the sign of the corresponding edge.

Property testing in the bounded-degree model is described as follows. Given oracle access to a graph G with degree bound d and $|V| = N$, we wish to distinguish whether the graph G satisfies a certain property, or whether it is ϵ-far from any graph having that property, where $\epsilon \in (0, 1]$ is an error parameter. Here we say that two graphs G and G' are ϵ-far from each other if we have to add or remove at least $\epsilon N d$ edges to turn G into G'. The specific case of clusterability testing is defined formally as follows.

▶ **Definition 1.** *A clusterability testing algorithm is a randomized algorithm that has query access to a signed graph $G(V, E, \Sigma)$ with $|V| = N$ and maximum degrees d. Given an error parameter ϵ, the algorithm behaves as follows:*

- *If G is clusterable, then the algorithm should accept with probability at least $2/3$.*
- *If G is ϵ-far from clusterable, then the algorithm rejects with probability at least $2/3$.*

3 Main Results and Proofs

In this section, we give the formal statements and proofs of our two main results – a classical query lower bound for clusterability testing and a quantum clusterability tester. In Section 3.1, we first give the classical query lower bound of $\Omega(\sqrt{N})$ for clusterability testing. This result claims the optimality of the classical clusterability tester in [1]. In Section 3.2, we provide a quantum clusterability tester with query complexity $\tilde{O}(N^{1/3})$ which outperforms the classical clusterability tester in [1].

3.1 Classical query lower bound for testing clusterability

In this section, we derive a classical query lower bound for the clusterability testing problem. Specifically, we show that testing the clusterability of a signed graph with N vertices requires at least $\sqrt{N}/10$ queries.

▶ **Theorem 2.** *Given a signed graph G with N vertices, testing clusterability of G with error parameter $\epsilon = 0.01$ requires at least $\sqrt{N}/10$ queries.*

Proof. The proof consists of three main steps. First, we construct two families of graphs denoted as $\mathcal{G}_1^N$ and $\mathcal{G}_2^N$, each possessing specific desirable properties. In particular, we require that most graphs within $\mathcal{G}_1^N$ are at least 0.01-far from being clusterable, while graphs within $\mathcal{G}_2^N$ are inherently clusterable. The construction and analysis of these families is deferred to Section 4.1.

To prove Theorem 2, we illustrate the interaction between an arbitrary T-query clusterability testing algorithm $\mathcal{A}$ and a graph g uniformly sampled from $\mathcal{G}_\alpha^N$ as follows:

For all $t \le T$, each query q_t is represented as a tuple (v_t, i_t), and the answer to q_t is denoted as a_t, where $v_t, a_t \in [N]$ and $i_t \in [6]$. It is crucial to note that each query q_t corresponds to an edge in g, specifically the edge (v_t, a_t). We additionally denote a list of pairs $h = [(q_1, a_1), (q_2, a_2), \ldots, (q_t, a_t)]$ as the query-answer history. This history is generated by the interaction between $\mathcal{A}$ and g in the following manner: For each $t \le T$, $\mathcal{A}$ maps h to q_{t+1} and ultimately to either accept or reject for $t = T$. For a given history $h = [(q_1 = (v_1, i_1), a_1), \ldots, (q_t = (v_t, i_t), a_t)]$, we say that a vertex u is in h if $u = v_{t'}$ or $u = a_{t'}$ for some $t' \in [t]$.

Secondly, we introduce two processes, denoted as P_α for $\alpha \in \{1, 2\}$, which simulate how an algorithm $\mathcal{A}$ interacts with a graph sampled uniformly from $\mathcal{G}_\alpha^N$. To be more specific, we consider that $\mathcal{A}$ interacts with a graph g sampled from $\mathcal{G}_\alpha^N$ and generates the query-answer history h. We must have that the graph g is uniformly distributed in $\mathcal{G}_{\alpha,h}^N \subseteq \mathcal{G}_\alpha^N$, where $\mathcal{G}_{\alpha,h}^N$ includes all graphs that produce the query-answer history h during interactions with $\mathcal{A}$. Therefore, if $\mathcal{A}$ makes a query $q_{t+1} \notin \{q_i\}_{i=1}^t$ to a graph uniformly sampled from $\mathcal{G}_{\alpha,h}^N$, we can determine that the answer corresponds to a certain vertex $u \in [N]$ with a specific probability denoted as p_u. The random processes P_α are precisely defined to return the answer u with the corresponding probability p_u when responding to the query q_{t+1} (initiated by $\mathcal{A}$) and considering the history h. As a result, these two random processes, P_α, interact with $\mathcal{A}$, providing responses to $\mathcal{A}$'s queries while simultaneously constructing a graph uniformly distributed in $\mathcal{G}_\alpha^N$. The description and analysis of these random processes are deferred to Section 4.2.

In the third part, we demonstrate that no algorithm can with high probability differentiate between query-answer histories generated during the interactions of $\mathcal{A}$ with P_1 and P_2 while making less than $\sqrt{N}/10$ queries. To prove such indistinguishability, we examine the distribution of query-answer histories of length T denoted as $\mathbf{D}_\alpha^{\mathcal{A}}$, where each element in $\mathbf{D}_\alpha^{\mathcal{A}}$ is generated through the interactions of $\mathcal{A}$ and P_α. The statistical difference between $\mathbf{D}_1^{\mathcal{A}}$ and $\mathbf{D}_2^{\mathcal{A}}$ is defined as follows:

$$\frac{1}{2} \cdot \sum_x \left| \mathrm{Prob}\left[\mathbf{D}_1^{\mathcal{A}} = x\right] - \mathrm{Prob}\left[\mathbf{D}_2^{\mathcal{A}} = x\right] \right|,$$

where x is some query-answer history of length T. We then provide an upper bound on this statistical difference in the following lemma. The proof of this lemma is a modification of the proof of Lemma 7.4 in [26], and we defer its proof to Section 4.3.

▶ **Lemma 3** (based on [26], Lemma 7.4)). *Let $\delta < \frac{1}{2}, T \le \delta\sqrt{N}$ and $N \ge 40T$. For every algorithm $\mathcal{A}$ that uses T queries, the statistical distance between $\mathbf{D}_1^{\mathcal{A}}$ and $\mathbf{D}_2^{\mathcal{A}}$ is at most $10\delta^2$.*

Finally, we establish Theorem 2 through a proof by contradiction. Let us assume the existence of a clusterability tester $\mathcal{A}$ that requires only $\sqrt{N}/10$ queries. Consequently, we can infer that the probability of $\mathcal{A}$ accepting a graph from $\mathcal{G}_2^N$ is at least 2/3. By referring to Lemma 3, we determine that the statistical difference between $\mathbf{D}_1^{\mathcal{A}}$ and $\mathbf{D}_2^{\mathcal{A}}$ is at most $10\delta^2 = 1/10$ where δ is set $1/10$ for a $\sqrt{N}/10$-query algorithm. Hence, $\mathcal{A}$ accepts a graph distributed uniformly in $\mathcal{G}_1^N$ with a probability of at least $2/3 - 1/10 > 0.4$.

Furthermore, as indicated by Proposition 7, more than 99% of the graphs in $\mathcal{G}_1^N$ are at least 0.01-far from being clusterable. Consequently, by the definition of a clusterability tester, we can conclude that $\mathcal{A}$ accepts a graph distributed uniformly in $\mathcal{G}_1^N$ with a probability of at most $0.99 \cdot 1/3 + 0.01 < 0.35$. This contradicts the earlier finding that $\mathcal{A}$ accepts a graph distributed uniformly in $\mathcal{G}_1^N$ with a probability of at least 0.4. Hence, we can deduce that there does not exist a clusterability tester capable of distinguishing between a graph sampled from $\mathcal{G}_1^N$ and $\mathcal{G}_2^N$ using only $\sqrt{N}/10$ queries, and the theorem follows. ◀

3.2 Quantum clusterability tester

Algorithm 1 Quantum clusterability tester.

```
Input: Oracle access to a signed graph $G(V, E, \Sigma)$ with $N$ vertices and degree bound $d$; an
    accuracy parameter $\epsilon \in (0, 1]$.
for $O(1/\epsilon)$ times do
    Pick a vertex $s \in V$ randomly.
    Let $K = O\left(\sqrt{N}\,\mathrm{poly}(\log N/\epsilon)\right)$, $L = \mathrm{poly}\,(\log N/\epsilon)$, $n = KL$, and $k = \Theta(L)$.
    Adopt Proposition 4 to construct $k$-wise independent random variables $b_{ij}$ taking
    values in $[2d]$ for $i \in [K]$ and $j \in [L]$.
    Run the quantum collision finding algorithm in Lemma 5 with the following setting:
      - $X := [K] \times [L]$; $Y$ is the set of pairs $(v, v_{\mathrm{neb}})$ where $v \in V$ and $v_{\mathrm{neb}}$ is the set of
        vertices adjacent to $v$.
      - A function $f$ that takes $(i, j) \in X$ as input, and returns the endpoint of a random
        walk that starts at $s$ with random coin flips $(b_{i1}, \ldots, b_{ij})$.
      - Symmetric binary relation $R \subseteq Y \times Y$ defined as follows:

          $((v, v_{\mathrm{neb}}), (v', v'_{\mathrm{neb}})) \in R$ iff ($v \in v'_{\mathrm{neb}}$ and the edge between $v$ and $v'$ is negative).

    if quantum collision finding algorithm finds a collision then
        return false
    end if
end for
return true
```

In this section, we present our second result: a quantum clusterability tester (Algorithm 1) with a query complexity of $O\left(N^{1/3}\,\mathrm{poly}\,(\log N/\epsilon)\right)$. We begin by introducing the quantum clusterability tester, followed by the proof of its correctness in Theorem 6.

Algorithm 1 takes a signed graph $G(V, E, \Sigma)$ with N vertices and a bound on the maximum degree d, along with an accuracy parameter $\epsilon \in (0, 1]$, as input. The goal is to determine whether $G(V, E, \Sigma)$ is clusterable or ϵ-far from clusterable. The algorithm consists of four major steps.

First, Algorithm 1 randomly selects a vertex $s \in V$. Second, it constructs random variables that determine the direction of movement in each step of these random walks. To achieve this, we need to prepare $O(K \cdot L)$ random variables (K and L are defined in Algorithm 1); however, we can derandomize and reduce the number of random bits from $O(K \cdot L)$ to $O(L)$ because Algorithm 1 only depends on each pair of walks that are selected from K random walks. Therefore, it is sufficient to construct k-wise independent random variables [1] b_{ij} mapping to $[2d]$ for $i \in [K]$ and $j \in [L]$, where $k = \Theta(L)$. This construction can be realized by the following proposition.

[1] A set of random variables is k-wise independent if any subset of k variables is independent.

▶ **Proposition 4** ([4], Proposition 6.5). *Let $n+1$ be a power of 2 and k be an odd integer such that $k \leq n$. In this scenario, there exists a uniform probability space denoted as $\Omega = \{0,1\}^m$, where $m = 1 + \frac{1}{2}(k-1)\log_2(n+1)$. Within this space, there exist k-wise independent random variables, represented as $\xi_1, \ldots, \xi_n$ over Ω, such that $\Pr[\xi_j = 1] = \Pr[\xi_j = 0] = \frac{1}{2}$.*

Moreover, an algorithm exists that, when provided with $i \in \Omega$ and $1 \leq j \leq n$, can compute $\xi_j(i)$ in a computational time of $O(k \log n)$.

Third, we define a function f that implements random walks according to these random variables. f returns the endpoint of a random walk and the neighborhood of this endpoint. Specifically, we let $X = \{1, \ldots, K\} \times \{1, \ldots, L\}$, and Y be the set of pairs (v, v_{neb}) where $v \in \{1, \ldots, N\}$ and v_{neb} is the set of vertices adjacent to v. Then, we define the function f as follows. f takes $(i,j) \in X$ as input, then it runs a random walk according to the random variables $(b_{i1}, \ldots, b_{ij})$ such that (i) this walk starts at s and (ii) each edge in this walk is positive. The function f finally returns (v, v_{neb}). Fourth, we define the symmetric binary relation $R \subseteq Y \times Y$ such that $((v, v_{\text{neb}}), (v', v'_{\text{neb}})) \in R$ iff (i) $v \in v'_{\text{neb}}$, and (ii) the edge between v and v' is negative. In other words, detecting a collision is equivalent to detecting a bad cycle. The last step is to detect two distinct elements $x_1, x_2 \in X$ such that $(f(x_1), f(x_2))$ satisfies the symmetric binary relation R. The collision finding problem can be improved by a quantum collision finding algorithm proposed by Ambainis et al. [11] as follow.

▶ **Lemma 5** ([11], Theorem 9). *Given a function $f : X \to Y$, and a symmetric binary relation $R \subseteq Y \times Y$ which can be computed in $\operatorname{poly}(\log |Y|)$ time steps where X and Y are some finite sets, we denote a collision by a distinct pair $x, x' \in X$ such that $(f(x), f(x')) \in R$. There exists a quantum algorithm that can find a collision with a constant probability when a collision exists, and always returns false when there does not exist a collision. The running time of the quantum algorithm is $O\left(|X|^{2/3} \cdot \operatorname{poly}(\log |Y|)\right)$.*

By this lemma we can identify a bad cycle within K random walks, with a query complexity of $O(|X|^{2/3}) = O((K \cdot L)^{2/3}) = O\left((\sqrt{N} \operatorname{poly}(\log N/\epsilon))^{2/3}\right) = O(N^{1/3} \operatorname{poly}(\log N/\epsilon))$. Next, we establish the correctness of this algorithm and present its time complexity in the following theorem.

▶ **Theorem 6.** *Algorithm 1 is a quantum algorithm that tests the clusterability of a signed graph with query complexity and running time $O(N^{1/3} \operatorname{poly}(\log N/\epsilon))$.*

Following our first result, we conclude that our quantum clusterability tester outperforms any classical clusterability tester.

Proof. First, we prove that Algorithm 1 is indeed a clusterability tester. When G is clusterable, signifying the absence of one bad cycle, Algorithm 1 fails to discover a collision. Consequently, it returns true. On the contrary, when G is ϵ-far from clusterable, the assertion in Claim 14 from [1] suggests that the algorithm can pinpoint a bad cycle within the sampled random walks with a constant probability. This leads Algorithm 1 to return false with a constant probability.

To bound the time complexity (and hence query complexity), we need to bound the following quantities:

- The time required to evaluate the k-wise independent random variables.
- The time required to evaluate f.
- The number of queries required to find a collision.

For the first requirement, it takes $O\left(\operatorname{poly}(\log N/\epsilon)\right)$ time to evaluate a k-wise independent random variable, as indicated by Proposition 4. Moving to the second requirement, it is evident that each evaluation of f consumes time $\operatorname{poly}(\log N/\epsilon)$ since f is a procedure implementing a random walk, and the length of the walk is $L \in \operatorname{poly}(\log N/\epsilon)$. Concerning the last requirement, we are aware that detecting a collision requires $O\left(N^{1/3} \operatorname{poly}\left(\log N/\epsilon\right)\right)$ time, as derived from Lemma 5. In conclusion, the query and time complexity of Algorithm 1 is $O\left(N^{1/3} \operatorname{poly}\left(\log N/\epsilon\right)\right)$. ◀

4 Proof details

In this section, we detail the construction and lemmas in Theorem 2. In Section 4.1, we generate two distinct families of graphs, each exhibiting different property of clusterability. In Section 4.2, we introduce two random processes that interact with an arbitrary algorithm $\mathcal{A}$ during the generation of graphs selected uniformly from the aforementioned families. In Section 4.3, we demonstrate that the statistical difference of query answer histories produced by $\mathcal{A}$ and these two random processes is bounded by the number of queries.

4.1 Graph construction

Here we detail the construction and analysis of the graph families $\mathcal{G}_1^N$ and $\mathcal{G}_2^N$.

4.1.1 Construction of two families of signed graphs

We detail the construction of two families of signed graphs denoted as $\mathcal{G}_1^N$ and $\mathcal{G}_2^N$. In both families, each signed graph consists of N vertices, where N is a multiple of 10. Each vertex v is assigned a label p_v chosen from the set $\{0, 1, \ldots, 9\}$ in such a way that there are exactly $N/10$ vertices for each possible label.

For the edge set, we embed them in a manner such that each vertex is incident to precisely 6 edges. We construct edge sets based on cycles associated to a permutation $\sigma = (r_1\ r_2\ \ldots\ r_L)$, where $0 \leq L \leq 9$ and $r_l \in \{0, 1, \ldots, 9\}$ are distinct for $1 \leq l \leq L$. With some abuse of notation, we also denote by σ the bijective function $\sigma : \{r_1, r_2, \ldots, r_L\} \to \{r_1, r_2, \ldots, r_L\}$ defined as

$$\sigma(r_l) = \begin{cases} r_{l+1} & \text{if } l < L. \\ r_1 & \text{if } l = L. \end{cases}$$

With this notation, we define a family $\mathcal{D}^\sigma$ such that each member of this family is a union of cycles satisfying two properties: (i) the union of cycles contains all vertices in $[N]$ labeled with values from r_0 to r_L, and (ii) for each cycle $(v_1, v_2, \ldots, v_J)$ in the union of cycles, the label for each vertex must satisfy $p_{v_{j+1}} = \sigma(p_{v_j})$ for $1 \leq j \leq J$ (where we set $v_{J+1} = v_1$). We then employ these cycles to define the edge sets for the graphs in the family $\mathcal{G}_\alpha^N$. See Figure 1 for an illustration for $\mathcal{G}_1^{40}$.

For $\mathcal{G}_1^N$: Each graph in $\mathcal{G}_1^N$ consists of one Hamiltonian cycle and two unions of cycles (we later comment on the particular choice of σ's):

- The Hamiltonian cycle $\in \mathcal{D}^{\sigma^{1\text{st}}}$ with $\sigma^{1\text{st}} = (0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9)$. We call this the arc cycle. All of its edges are positive, and we refer to these edges as arc edges.

- One union of cycles $\in \mathcal{D}^{\sigma^{2\text{nd}}}$ with $\sigma^{2\text{nd}} = (2\ 4\ 6\ 0\ 8\ 1\ 3\ 7\ 9\ 5)$,[2] with each of its edges being positively signed. We call these edges connecting edges.
- A second union of cycles $\in \mathcal{D}^{\sigma^{3\text{rd}}}$ with $\sigma^{3\text{rd}} = (1\ 6\ 3\ 8\ 5\ 0\ 7\ 2\ 9\ 4)$, and all edges are negatively signed.

For $\mathcal{G}_2^N$: In the family of graphs $\mathcal{G}_2^N$, each graph contains one Hamiltonian cycle and 12 unions of cycles:

- The Hamiltonian cycle $\in \mathcal{D}^{\sigma^{1\text{st}}}$ with each of its edges negatively signed.
- There are ten additional unions of cycles $\in \mathcal{D}^{\sigma^s}$ with $\sigma^s = (s)$ for s taking values in the set $\{0, 1, \dots, 9\}$. These edges are positive.
- The last two unions of cycles are disjoint. One belongs to $\mathcal{D}^{\sigma^{10}}$ with $\sigma^{10} = (0\ 2\ 4\ 6\ 8)$. The other belongs to $\mathcal{D}^{\sigma^{11}}$ with $\sigma^{11} = (1\ 3\ 5\ 7\ 9)$. These edges are positive.

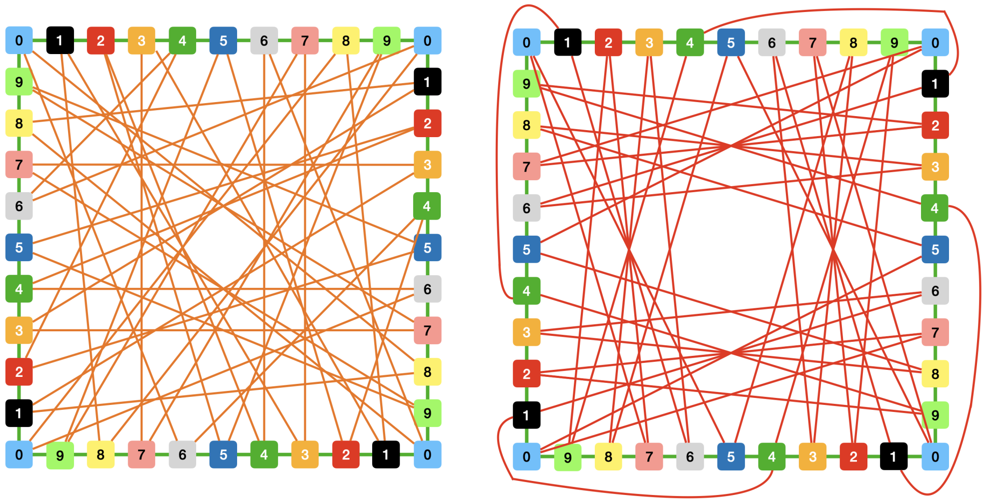


Figure 1 An instance of $\mathcal{G}_1^{40}$. The green lines indicate the edges in one Hamiltonian cycle belonging to $\mathcal{D}^{\sigma^{1\text{st}}}$, the orange lines indicate the edges in one union of cycles belonging to $\mathcal{D}^{\sigma^{2\text{nd}}}$, and the red lines indicate the edges in one union of cycles belonging to $\mathcal{D}^{\sigma^{3\text{rd}}}$.

In every graph within $\mathcal{G}_\alpha^N$, each vertex is incident to precisely six edges, and these edges are labeled according to the following convention: For a pair of adjacent vertices, represented as v_j and v_{j+1} for $1 \le j \le J-1$, within any cycle $(v_1, v_2, \dots, v_J)$, we label the edge connecting them as k for v_m and as $k+1$ for v_{m+1}, for some $k \in \mathbb{N}$. This labeling effectively associates an orientation to the cycle. More specifically, in a graph from $\mathcal{G}_1^N$:

- The edges in the Hamiltonian cycle from $\mathcal{D}^{\sigma^{1\text{st}}}$ are labeled with 1 and 2.
- For the edges in the union of cycles $\in \mathcal{D}^{\sigma^{2\text{nd}}}$, we use labels 3 and 4.
- For the edges in the union of cycles $\in \mathcal{D}^{\sigma^{3\text{rd}}}$, we use labels 5 and 6.

In the case of a graph from $\mathcal{G}_2^N$:

- The edges in the Hamiltonian cycle corresponding to $\mathcal{D}^{\sigma^{1\text{st}}}$ are labeled as 5 and 6.

[2] The choice of $\sigma^{2\text{nd}}$ and $\sigma^{3\text{rd}}$ is not unique; we only require that these edge sets are disjoint when we fix the label of each vertex. This forbids picking for example $\sigma^{2\text{nd}} = (0\ 2\ 4\ 6\ 8\ 1\ 3\ 5\ 7\ 9)$, since the edges connecting vertices labeled 9 and 0 can be found in both $\mathcal{D}^{\sigma^{1\text{st}}}$ and $\mathcal{D}^{\sigma^{2\text{nd}}}$, meaning they are not disjoint. However, we could replace $\sigma^{2\text{nd}}$ with $(2\ 6\ 4\ 0\ 8\ 1\ 3\ 7\ 9\ 5)$, where exchanging 6 and 4 would not violate the disjoint property.

- For the edges in the union of cycles within $\mathcal{D}^{\sigma^s}$ for s ranging from 0 to 9, we assign labels 1 and 2.
- For the edges in the union of cycles $\in \mathcal{D}^{\sigma^{10}}$ and $\mathcal{D}^{\sigma^{11}}$, we label them with 3 and 4.

4.1.2 Clusterability of two families of signed graphs

We initially observe that all graphs within $\mathcal{G}_2^N$ are clusterable with the following rationale. All vertices with even (odd, respectively) labeling are interconnected via positive edges. Consequently, two connected components emerge: one component comprises all vertices with even labeling, and the other includes all vertices with odd labeling, each with positive edges. We further note that these two components can only be connected through negative edges. Hence, any graph within $\mathcal{G}_2^N$ satisfies the clusterability definition. As a result, all graphs in the second family are clusterable.

Regarding the graphs in $\mathcal{G}_1^N$, we will demonstrate that they are at least 0.01-far from being clusterable with probability at least $1 - \exp(-\Omega(N))$ in the following Proposition 7.

▶ **Proposition 7.** *The graphs in $\mathcal{G}_1^N$ are* 0.01*-far from clusterable with probability at least* $1 - \exp(-\Omega(N))$.

Proof. We commence our proof by providing a description of the random process used to uniformly generate a graph denoted as g from $\mathcal{G}_1^N$. We begin by constructing the set of vertices $[N]$ and refer to the resulting (empty) graph as sg. The graph sg is equipped with its edges set through a three-step process:

1. **(One Hamiltonian cycle):** In the first step, we uniformly select a Hamiltonian cycle from all possible Hamiltonian cycles on the vertices set $[N]$ and assign each vertex a label from the set $\{0, 1, \ldots, 9\}$, based on the rule of cycles $\in \mathcal{D}^{\sigma^{1\text{st}}}$. All edges constructed in this step are positive.
2. **(Second edge set from $\mathcal{D}^{\sigma^{2\text{nd}}}$):** In the second step, we repeat the following processes N times:
   a. Select an arbitrary vertex u_i that lacks an edge labeled as 3 where the index $i \in [N]$ represents the label of iteration.
   b. Uniformly select a vertex v_i from a set that includes all vertices labeled as $\sigma^{2\text{nd}}(p_{u_i})$ and that lack an edge labeled as 4.
   c. Add the edge (u_i, v_i).

   This adds an edge set from $\mathcal{D}^{\sigma^{2\text{nd}}}$. We make these edges positive.
3. **(Third edge set from $\mathcal{D}^{\sigma^{3\text{rd}}}$):** Similar to the previous procedure, we add an edge set from $\mathcal{D}^{\sigma^{3\text{rd}}}$. We make these edges negative.

We call the resulting graph g, and note that g is a uniformly random element from $\mathcal{G}_1^N$. We proceed to observe that each graph g in $\mathcal{G}_1^N$ is inherently non-clusterable. Indeed, unless we remove arc edges from the Hamiltonian cycle, every negative edge of a cycle in $\mathcal{D}^{\sigma^{3\text{rd}}}$ contributes to a bad cycle. We show that, with high probability over the random graph in $\mathcal{G}_1^N$, removing less than $0.01dN = 0.06N$ edges cannot make the graph clusterable.

More precisely, we will establish that after removing less than $0.06N$ arc edges, with high probability, all vertices remain connected through (positive) connecting edges in $\mathcal{D}^{\sigma^{2\text{nd}}}$, and so the graph cannot be clustered. To prove this, let us delve into a more detailed description of the random process used to generate a graph g.

In the first step, we construct a Hamiltonian cycle and eliminate $x < 0.06N$ arc edges, resulting in a graph with x components. There are $C_x^N = \binom{N}{x}$ possible possibilities for these x components.

During the first iteration of the second step, we select the arbitrary vertex u_1 from the component with the fewest vertices and designate this component as C. It becomes evident that, in the first iteration of step 2(c), the edge (u_1, v_1) connects the component C to another component, with a probability exceeding $1/2$. Consequently, the number of components in the graph sg decreases by 1, and the number of vertices in C increases with a probability greater than $1/2$.

In the subsequent iterations, we select the vertex u_i for $2 \leq i \leq N$ based on the following rule: If the number of vertices labeled as $\sigma^{2\text{nd}}(p_{v_{i-1}})$ and lacking edges labeled as 4 within the component C is fewer than the number of vertices labeled as $\sigma^{2\text{nd}}(p_{v_{i-1}})$ not in C, then we set the vertex u_i equal to v_{i-1}. Subsequently, in 2(b) and 2(c), the process embeds an edge connecting u_i to some vertex v_i that is not a resident in C with a probability greater than $1/2$. Otherwise, we choose u_i from any arbitrary vertex labeled as $\sigma^{2\text{nd}}(p_{v_{i-1}})$ and not in C. Subsequently, in 2(b), the process selects v_i in C with a probability greater than $1/2$, as C has more vertices capable of connecting with u_i than the set of vertices not in C.

Consequently, the probability of the graph having more than one component can be bounded by the probability of obtaining fewer than x heads when flipping N unbiased coins. This probability can be bounded as follows:

$$\sum_{i=2}^{x} C_i^N \left(\frac{1}{2}\right)^{N-i} < 2^{N \cdot H(0.06)} 2^{-N} 2^{x} < 2^{N(-1+0.06+H(0.06))},$$

where $H(p) = -p\log(p) - (1-p)\log(1-p)$ is the (binary) entropy function.

At this point, we removed only $x < 0.06N$ edges, and we are permitted to remove additional $0.06N - x$ connecting edges from sg. This corresponds to $0.06N - x$ tests where the coin flips tails (thus reducing the number of components by 1) can be taken into account for flips resulting in heads. In other words, the condition of having fewer than x heads can be extended to having fewer than $x + 0.06N - x$ heads when flipping N unbiased coins. Consequently, the probability that the resulting graph has more than one component, when $0.06N - x$ connecting edges are removed, can be bounded as:

$$\sum_{i=2}^{x+(0.06N-x)} C_i^N \left(\frac{1}{2}\right)^{N-i} < 2^{N(-1+0.06+H(0.06))}.$$

Given that there are $C_x^N < 2^{NH(0.06)}$ possible ways to construct x components in the first step, we can confidently assert that, after implementing step (2), all vertices in sg are interconnected by positive edges with a probability of at least $1 - \exp^{-\Omega(N)}$, even in cases where $0.06N$ positive edges (comprising x arc edges and $0.06N - x$ connecting edges) were removed. The negative edges are present in the edge set in $\mathcal{D}^{\sigma^{3\text{rd}}}$, and each of them generates a bad cycle under the condition that only one component (only positive edges inside) is left after completing the second step in the process. In other words, under this condition, we must remove all negative edges to make this graph clusterable. Consequently, the lemma follows. ◀

4.2 Random processes

Here, we construct and analyze the random processes that play a key role in our lower bound. The first part describes the interaction of a random process P_α with an algorithm $\mathcal{A}$. The second part proves that P_α indeed generates a graph uniformly within $\mathcal{G}_\alpha^N$, as further elucidated in Proposition 8.

We will begin by defining the random process P_1, which involves two stages. The first stage will explain how P_1 interacts with an arbitrary T-query algorithm $\mathcal{A}$. The second stage will elaborate on how P_1 constructs a graph uniformly sampled from $\mathcal{G}_1^N$.

First stage of P_1: Given a query-answer history $h = [(q_1, a_1), (q_2, a_2), \ldots, (q_{t-1}, a_{t-1})]$ for $t \le T$, we define a set of vertices $X_{p,i}$, which contains all vertices labeled with p in the history and lacking edge labeled i. We also use the notation n_p to represent the count of vertices in this history that are labeled p. For each query $q_t = (v_t, i_t)$ made by $\mathcal{A}$, the actions of P_1 are defined as follows:

1. If v_t is not in h, then P_1 labels v_t with a number $p \in \{0, 1, \cdots, 9\}$ with a probability of $\frac{(N/10)-n_p}{N-\left(\sum_{p=0}^{9} n_p\right)}$. Subsequently, P_1 answers this query as described in (2) below.
2. If v_t belongs to h, there are two possible scenarios:
   a. If we can find the edge corresponding to q_t in h, then P_1 responds with the vertex connected to this edge. In other words, there exists an edge (v_t, u) in h such that (v_t, u) is labeled as i_t for vertex v_t, and P_1 responds with u. The query-answer history remains unchanged in this case.
   b. If the edge corresponding to $q_t = (v_t, i_t)$ does not exist in h, we follow these steps: Suppose, without loss of generality, that $i_t = 1$. We set the label $\sigma^{\text{1st}}(p_{v_t})$ as $\overline{p}$ and $\overline{i} = i_t + 1 = 2$. P_1 decides whether to uniformly select a vertex from $X_{\overline{p},\overline{i}}$ by flipping a coin with bias $\frac{|X_{\overline{p},\overline{i}}|}{N/10 - n_{\overline{p}} + |X_{\overline{p},\overline{i}}|}$ or to uniformly select a vertex not present in h, and assigns the label $\overline{p}$ to it. In either case, P_1 responds with the selected vertex u, and the edge (v_t, u) is signed positively. Subsequently, this edge (v_t, u) is added to the query-answer history h.
   For the other case ($i_t = 2, 3, 4, 5, 6$), P_1 acts similarly as described above, except for the assignment for $\overline{p}$, the assignment for $\overline{i}$, and the sign of the added edge. The added edge is positively signed for $i_t = 2, 3, 4$, and negatively signed for $i_t = 5, 6$. For $\overline{i}$, it is set to $i_t + 1$ for $i_t = 3, 5$ and to $i_t - 1$ for $i_t = 2, 4, 6$. The assignment for $\overline{p}$ is as follows:

$$\begin{cases} \overline{p} \leftarrow (\sigma^{\text{1st}})^{-1}(p_{v_t}) & \text{for } i_t = 2 \\ \overline{p} \leftarrow \sigma^{\text{2nd}}(p_{v_t}) & \text{for } i_t = 3 \\ \overline{p} \leftarrow (\sigma^{\text{2nd}})^{-1}(p_{v_t}) & \text{for } i_t = 4 \\ \overline{p} \leftarrow \sigma^{\text{3rd}}(p_{v_t}) & \text{for } i_t = 5 \\ \overline{p} \leftarrow (\sigma^{\text{3rd}})^{-1}(p_{v_t}) & \text{for } i_t = 6 \end{cases}$$

First stage of P_2: P_2 follows a similar process to P_1, with the only differences being the assignment for $\overline{p}$ as follows:

$$\begin{cases} \overline{p} \leftarrow p_{v_t} & \text{for } i_t = 1 \\ \overline{p} \leftarrow p_{v_t} & \text{for } i_t = 2 \\ \overline{p} \leftarrow p_{v_t} + 2 \pmod{10} & \text{for } i_t = 3 \\ \overline{p} \leftarrow p_{v_t} - 2 \pmod{10} & \text{for } i_t = 4 \\ \overline{p} \leftarrow \sigma^{\text{1st}}(p_{v_t}) & \text{for } i_t = 5 \\ \overline{p} \leftarrow (\sigma^{\text{1st}})^{-1}(p_{v_t}) & \text{for } i_t = 6 \end{cases}$$

Second stage of P_1: After answering all of these queries and generating a query-answer history $[(q_1, a_1), \ldots, (q_T, a_T)]$, P_1 proceeds with the following processes:

1. Uniformly selecting a feasible way to embed the edges in h on a cycle. The embedding of these edges adhere to the following conditions: Each vertex is assigned a cycle position, i.e., an integer in $\{0, \ldots, N-1\}$, in a manner that ensures each vertex labeled with $p \in \{0, \ldots, 9\}$ is positioned at a position x such that $p \equiv x \pmod{10}$. This assignment implies that all acr edges (labeled $1, 2$) are placed on the cycle, and edges labeled $3, 4, 5, 6$ are excluded from the cycle.
2. Randomly positioning all other vertices on the cycle, ensuring that each vertex v with label p_v is positioned at a position x such that $p_v \equiv x \pmod{10}$. Subsequently, all cycle edges are assigned a positive sign.
3. In the end, uniformly selecting a feasible way to embed the edges sets in $\mathcal{D}^{\sigma^{2\text{nd}}}$ and $\mathcal{D}^{\sigma^{3\text{rd}}}$. All edges in the edges sets $\in \mathcal{D}^{\sigma^{2\text{nd}}}$ assigned a positive sign, while all edges in the edges sets $\in \mathcal{D}^{\sigma^{3\text{rd}}}$ are assigned a negative sign.

Second stage of P_2: P_2 follows a process similar to that of P_1, with few distinctions: In (2), we assign each cycle edge a negative sign. In (3), P_2 uniformly selects a feasible way to embed the edges sets $\in \mathcal{D}^{\sigma^s}$ for $s \in \{0, 1, \cdots, 11\}$, and assigns positive signs to these edges.

We will show that the above two processes uniformly generate a graph in the corresponding family in the next lemma.

▶ **Proposition 8.** *For every algorithm $\mathcal{A}$ that uses T queries and for each $\alpha \in \{1, 2\}$, the process P_α uniformly generates graphs in $\mathcal{G}_\alpha^N$ when interacting with $\mathcal{A}$.*

Proof. We will use induction to prove this lemma. Consider that every probabilistic algorithm can be viewed as a distribution of deterministic algorithms. Therefore, it is sufficient to prove this lemma for any deterministic algorithm $\mathcal{A}$. The base case (i.e., $T = 0$) is correct because the query-answer history is empty, and the second stage in the process P_α uniformly generates a graph in $\mathcal{G}_\alpha^N$. We assume that the claim is true for $T-1$, and we will prove that the claim is also true for T. Let $\mathcal{A}'$ be the algorithm defined by stopping $\mathcal{A}$ before it asks the T^{th} query. By the inductive assumption, we know that P_α uniformly generates graphs in $\mathcal{G}_\alpha^N$ when P_α interacts with $\mathcal{A}'$. We will show that after P_α interacts with $\mathcal{A}$ and answers the T^{th} query, the second stage of P_α also uniformly generates graphs in $\mathcal{G}_\alpha^N$.

Assuming, without loss of generality, that the answer to the T^{th} query cannot be obtained from the query-answer history because this query does not provide additional information. Denote the T^{th} query of $\mathcal{A}$ as $q_T = (v_T, i_T)$ and consider all actions of the process P_1:

- **(Case 1) $\mathbf{i_T \in \{3, 4, 5, 6\}}$, and $\mathbf{v_T}$ in** h:
Assume, without loss of generality, that $i_T = 3$ and denote $\overline{p} = \sigma^{2\text{nd}}(p_{v_T})$. The probability of P_1 connecting v_T to any vertex is independent of the specific order of vertices on the cycle but depends on the labeling of the vertices. After considering all possible connecting edges carried out in the second stage following the interaction with $\mathcal{A}'$, it becomes evident that the only vertices in h to which v_T can connect are those in $X_{\overline{p},4}$. In any potential arrangement of the vertices on the cycle, there will be exactly $(N/10) - n_{\overline{p}}$ vertices labeled $\overline{p}$ and available for connection to v_T. This implies that the probability of v_T being connected to a vertex in $X_{\overline{p},4}$ is $\frac{|X_{\overline{p},4}|}{|X_{\overline{p},4}|+(N/10)-n_{\overline{p}}}$. Furthermore, when v_T is connected to a vertex in $X_{\overline{p},4}$, this vertex is uniformly distributed within $X_{\overline{p},4}$. Similarly, when connected to a vertex not in h, this vertex is uniformly distributed among the vertices not in h. These probabilities align with the definitions in P_1. Therefore, in Case 1, the induction step holds for P_1.

- **(Case 2) $\mathbf{i_T} \in \{\mathbf{1,2}\}$, and $\mathbf{v_T}$ in h:**
  Assume, without loss of generality, that $i_T = 1$ and denote $\overline{p}$ as $\sigma^{1\text{st}}(p_{v_T})$. In any valid embedding of the edges in h onto the cycle, it is evident that v_T can be adjacent with another vertex u in h only if u belongs to $X_{\overline{p},2}$. Moreover, when v_T is adjacent to a vertex in h, this vertex is uniformly distributed within $X_{\overline{p},2}$. If v_T is adjacency with another vertex u not in h, it is evident that the number of vertices labeled $\overline{p}$ but not in h is $(N/10) - n_{\overline{p}}$. Consequently, the probability of v_T being adjacent to some $u \in X_{\overline{p},2}$ is $\frac{|X_{\overline{p},2}|}{|X_{\overline{p},2}|+(N/10)-n_{\overline{p}}}$, and the probability of it being adjacent to a vertex not in h is $\frac{(N/10)-n_{\overline{p}}}{|X_{\overline{p},2}|+(N/10)-n_{\overline{p}}}$. These probabilities align with the definitions in P_1. Therefore, in this case, the induction step holds for P_1.
- **(Case 3) $\mathbf{v_T}$ is not in h:**
  We can reduce this case to case 1 and 2, provided that the label of v_T is selected with the appropriate probability. In the second stage, each vertex is randomly assigned label based on the proportion of missing vertices with that label. This essentially follows the assignment rule outlined in case (1) in the first stage of P_1.

For P_2, we omit the proof since it is similar to the argument in P_1, and this lemma follows. ◀

4.3 Proof of Lemma 3

We may assume that $\mathcal{A}$ does not make a query whose answer can be obtained from its query answer history h since such a query does not update the h. Then, we begin the proof by proving the following proposition.

▶ **Proposition 9.** *([26], Claim in lemma 7.4) Both in $\mathbf{D}_1^{\mathcal{A}}$ and in $\mathbf{D}_2^{\mathcal{A}}$, the total probability mass assigned to query-answer histories in which for some $t \leq T$ a vertex in h is returned as an answer to the t^{th} query is at most $10\delta^2$.*

Proof. We begin the proof by claiming that the probability of the event that the answer in the t^{th} query is a vertex in h is at most $20(t-1)/N$ for every $t \leq T$. The statement can be derived by observing that there are at most $2(t-1)$ vertices in h, and uses the definition of both processes. Then, the probability that the event occurs in an arbitrary query-answer history of length T is at most $\sum_{t=1}^{\delta\sqrt{N}} \frac{20(t-1)}{N} < 10\delta^2$. The proposition follows. ◀

From the proposition, we know that the edges in h will not form a cycle with probability at least $1 - 10\delta^2$. This event implies that for each query, these two processes pick a random vertex uniformly among the vertices, not in h. In addition, $\mathcal{A}$'s queries can only depend on the previous query-answer histories. Therefore, the distributions of the query-answer histories for these two processes are identical, except if we found a cycle, which happens with probability at most $10\delta^2$. Lemma 3 follows.

References

1 Florian Adriaens and Simon Apers. Testing cluster properties of signed graphs. In *Proceedings of the ACM Web Conference 2023*, pages 49–59, 2023.

2 Nir Ailon, Bernard Chazelle, Seshadhri Comandur, and Ding Liu. Estimating the distance to a monotone function. *Random Structures & Algorithms*, 31(3):371–383, 2007.

3 Noga Alon. Testing subgraphs in large graphs. *Random Structures & Algorithms*, 21(3-4):359–370, 2002.

4 Noga Alon, László Babai, and Alon Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *Journal of algorithms*, 7(4):567–583, 1986.

5 Noga Alon, Seannie Dar, Michal Parnas, and Dana Ron. Testing of clustering. *SIAM Journal on Discrete Mathematics*, 16(3):393–417, 2003.

6 Noga Alon, Tali Kaufman, Michael Krivelevich, and Dana Ron. Testing triangle-freeness in general graphs. *SIAM Journal on Discrete Mathematics*, 22(2):786–819, 2008.

7 Noga Alon and Michael Krivelevich. Testing k-colorability. *SIAM Journal on Discrete Mathematics*, 15(2):211–227, 2002.

8 Noga Alon and Asaf Shapira. Testing satisfiability. *Journal of Algorithms*, 47(2):87–103, 2003.

9 Noga Alon and Asaf Shapira. Every monotone graph property is testable. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 128–137, 2005.

10 Andris Ambainis, Aleksandrs Belovs, Oded Regev, and Ronald de Wolf. Efficient quantum algorithms for (gapped) group testing and junta testing. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 903–922. SIAM, 2016.

11 Andris Ambainis, Andrew M Childs, and Yi-Kai Liu. Quantum property testing for bounded-degree graphs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 365–376. Springer, 2011.

12 Simon Apers and Alain Sarlette. Quantum fast-forwarding: Markov chains and graph property testing. *arXiv preprint*, 2018. arXiv:1804.02321.

13 Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine learning*, 56(1):89–113, 2004.

14 Andrej Bogdanov, Kenji Obata, and Luca Trevisan. A lower bound for testing 3-colorability in bounded-degree graphs. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, pages 93–102. IEEE, 2002.

15 Sebastien Bubeck, Sitan Chen, and Jerry Li. Entanglement is necessary for optimal quantum property testing. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 692–703. IEEE, 2020.

16 Harry Buhrman, Lance Fortnow, Ilan Newman, and Hein Röhrig. Quantum property testing. *SIAM Journal on Computing*, 37(5):1387–1400, 2008.

17 Nilanjana Datta, Milan Mosonyi, Min-Hsiu Hsieh, and Fernando GSL Brandao. A smooth entropy approach to quantum hypothesis testing and the classical capacity of quantum channels. *IEEE transactions on information theory*, 59(12):8014–8026, 2013.

18 James A Davis. Clustering and structural balance in graphs. *Human relations*, 20(2):181–187, 1967.

19 Erik D Demaine, Dotan Emanuel, Amos Fiat, and Nicole Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2-3):172–187, 2006.

20 Ilias Diakonikolas, Homin K Lee, Kevin Matulef, Krzysztof Onak, Ronitt Rubinfeld, Rocco A Servedio, and Andrew Wan. Testing for concise representations. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 549–558. IEEE, 2007.

21 Irit Dinur, Shai Evra, Ron Livne, Alexander Lubotzky, and Shahar Mozes. Locally testable codes with constant rate, distance, and locality. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 357–374, 2022.

22 Eldar Fischer, Guy Kindler, Dana Ron, Shmuel Safra, and Alex Samorodnitsky. Testing juntas. *Journal of Computer and System Sciences*, 68(4):753–787, 2004.

23 Oded Goldreich. Short locally testable codes and proofs (survey). In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 14, 2005.

24 Oded Goldreich. Property testing. *Lecture Notes in Comput. Sci*, 6390, 2010.

25 Oded Goldreich, Shari Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM (JACM)*, 45(4):653–750, 1998.

26 Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 406–415, 1997.

27 Tom Gur, Min-Hsiu Hsieh, and Sathyawageeswar Subramanian. Sublinear quantum algorithms for estimating von neumann entropy. *arXiv preprint*, 2021. `arXiv:2111.11139`.

28 Frank Harary. On the notion of balance of a signed graph. *Michigan Mathematical Journal*, 2(2):143–146, 1953.

29 Charanjit S Jutla, Anindya C Patthak, Atri Rudra, and David Zuckerman. Testing low-degree polynomials over prime fields. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 423–432. IEEE, 2004.

30 Pieter W Kasteleyn. Dimer statistics and phase transitions. *Journal of Mathematical Physics*, 4(2):287–293, 1963.

31 Tali Kaufman and Dana Ron. Testing polynomials over general fields. *SIAM Journal on Computing*, 36(3):779–802, 2006.

32 Tali Kaufman and Madhu Sudan. Algebraic property testing: the role of invariance. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 403–412, 2008.

33 Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Signed networks in social media. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1361–1370, 2010.

34 Ting-Chun Lin and Min-Hsiu Hsieh. c 3-locally testable codes from lossless expanders. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 1175–1180. IEEE, 2022.

35 Ashley Montanaro and Ronald de Wolf. A survey of quantum property testing. *arXiv preprint*, 2013. `arXiv:1310.2035`.

36 Roland Nagy, Matthias Widmann, Matthias Niethammer, Durga BR Dasari, Ilja Gerhardt, Öney O Soykal, Marina Radulaski, Takeshi Ohshima, Jelena Vučković, Nguyen Tien Son, et al. Quantum properties of dichroic silicon vacancies in silicon carbide. *Physical Review Applied*, 9(3):034022, 2018.

37 Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical ldpc codes. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 375–388, 2022.

38 Dana Ron et al. Property testing: A learning theory perspective. *Foundations and Trends® in Machine Learning*, 1(3):307–402, 2008.

39 Dana Ron et al. Algorithmic and analysis techniques in property testing. *Foundations and Trends® in Theoretical Computer Science*, 5(2):73–205, 2010.

40 Jiliang Tang, Yi Chang, Charu Aggarwal, and Huan Liu. A survey of signed network mining in social media. *ACM Computing Surveys (CSUR)*, 49(3):1–37, 2016.

41 Luca Trevisan. Some applications of coding theory in computational complexity. *arXiv preprint*, 2004. `arXiv:cs/0409044`.

Quantum Non-Identical Mean Estimation: Efficient Algorithms and Fundamental Limits

Jiachen Hu
Peking University, Beijing, China

Tongyang Li
Peking University, Beijing, China

Xinzhao Wang
Peking University, Beijing, China

Yecheng Xue
Peking University, Beijing, China

Chenyi Zhang
Stanford University, CA, USA

Han Zhong
Peking University, Beijing, China

Abstract

We systematically investigate quantum algorithms and lower bounds for mean estimation given query access to non-identically distributed samples. On the one hand, we give quantum mean estimators with quadratic quantum speed-up given samples from different bounded or sub-Gaussian random variables. On the other hand, we prove that, in general, it is impossible for any quantum algorithm to achieve quadratic speed-up over the number of classical samples needed to estimate the mean μ, where the samples come from different random variables with mean close to μ. Technically, our quantum algorithms reduce bounded and sub-Gaussian random variables to the Bernoulli case, and use an uncomputation trick to overcome the challenge that direct amplitude estimation does not work with non-identical query access. Our quantum query lower bounds are established by simulating non-identical oracles by parallel oracles, and also by an adversarial method with non-identical oracles. Both results pave the way for proving quantum query lower bounds with non-identical oracles in general, which may be of independent interest.

2012 ACM Subject Classification Theory of computation → Quantum query complexity

Keywords and phrases Quantum algorithms, Mean estimation, Non-identical samples, Query complexity

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.9

Related Version *Full Version*: `https://arxiv.org/abs/2405.12838` [14]

Funding TL, XW, and YX were supported by the National Natural Science Foundation of China (Grant Numbers 62372006 and 92365117), and The Fundamental Research Funds for the Central Universities, Peking University.

1 Introduction

The problem of estimating the mean μ of a random variable X given its i.i.d. samples is a fundamental problem in statistics. For any random variable X with finite variance σ^2, the median-of-means estimator can estimate μ to within additive error ϵ with failure probability $\leq \delta$ using $O(\frac{\sigma^2}{\epsilon^2}\log(\frac{1}{\delta}))$ samples. This sample complexity is known to be tight up to a constant multiplicative factor [7].

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 9; pp. 9:1–9:21
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

On the other hand, suppose that a quantum computer has access to a unitary U and its inverse such that $U|\mathbf{0}\rangle$ encodes the random variable X coherently, and each application of U and $U^\dagger$ as a black-box oracle can be regarded as a quantum analogue of getting a sample of the random variable X. Therefore, the application of U is sometimes called a *quantum experiment* [11]. Under this assumption, a quantum computer can estimate the mean of X with $O(\frac{\sigma}{\epsilon}\log(\frac{1}{\delta}))$ quantum experiments [17], which achieves quadratic speed-up compared to the classical counterpart. Such quantum mean estimators embrace various applications, including approximate counting [17, 6], data stream estimation [12], derivative pricing in finance [5], etc.

In some cases, we are interested in estimating the mean of "close" random samples, such as random samples with the same mean but different distributions. For example, it is ubiquitous that the measurements of random samples have small systematic errors. In such cases there may be small difference between the means of the actual distributions of the measured random samples, and our algorithms and lower bounds also take this into account. One specific example is to learn a linear system discussed below. In classical mean estimation, the same method for identical random variables also works for non-identical random variables. As long as the variance of all random variables is bounded by σ^2, the median-of-means estimator can be directly adapted to these situations , yielding an algorithm with the same complexity. However, it is unclear whether similar results hold in the regime of quantum mean estimation. Therefore, it is a natural question whether we can achieve quantum speed-up for the mean estimation problem with non-identically distributed samples.

Below we provide a potential application for the quantum mean estimation with non-identically distributed samples.

Quantum Linear System

A classical linear dynamical system (LDS) is defined as

$$x_{t+1} = Ax_t + w_t,\ x_t \in \mathbb{R}^n,\ w_t \sim \mathcal{N}(\mathbf{0}, \sigma_w^2),\ \|A\|_2 < 1, x_1 = \mathbf{0} \tag{1}$$

where x_t is the state at time step t, and w_t is a random noise at step t. A well-known problem in LDS is to do the system identification: estimating the transition matrix A given a series of states starting from step 1. The standard approach to estimate transition matrix A in the classical linear system is ordinary least squares (OLS) [8, 20].

Consider the quantum counterpart of LDS (for example, when simulating a LDS on a quantum computer):

$$U_f|\psi_x\rangle|0\rangle = \int_{\mathbb{R}^n} \sqrt{f_w(w)}|\psi_x\rangle|\psi_{Ax+w}\rangle \mathrm{d}w, \tag{2}$$

$$U_o|\psi_x\rangle|0\rangle = |\psi_x\rangle|x\rangle, \tag{3}$$

here $f_w(w)$ is the probability density function (pdf) of $\mathcal{N}(\mathbf{0}, \sigma_w^2)$, and $|\psi_x\rangle$ is an arbitrary embedding of the raw state x. It is natural to ask whether it is possible to estimate A by a quantum algorithm with desired speed-up in quantum linear systems. Actually, it is indeed possible with a procedure presented in Section 4.1.3. This estimation procedure uses multiple calls to U_f to construct a new oracle U_{t_0} for some step t_0, which encodes a probability distribution over the matrix space with A as the mean value. However, the distribution encoded by U_{t_0} is different for different t_0, though their means are all equal to A. Therefore, this problem presents another motivation of the quantum non-identical mean estimation problem.

In general, the quantum linear system problem described above is a special class of quantum estimation problem in which quantum probability oracles have a time-varying zero-mean noise. The distribution of noise at each step is different but all zero-mean. The number of samples at each step is limited.

1.1 Contributions

In this paper, we systematically analyze the sample complexity of the *quantum non-identical mean estimation problem* (see its formal definition in Task 8). Roughly speaking, the quantum algorithm is given T different random variables in turn and can get $m \in \mathbb{N}$ samples from each random variable. Suppose that the mean of every random variable is in $(\mu - c\epsilon, \mu + c\epsilon)$ for some constant $0 < c < 1$, the quantum non-identical mean estimation problem is to estimate μ up to additive error ϵ. If all random variables are bounded or sub-Gaussian (see definition in Definition 14), for accuracy ϵ and $m = \Omega(\log(\frac{1}{\epsilon}))$, we give quantum algorithms solving the quantum non-identical mean estimation problem with quadratic speed-up.

▶ **Theorem 1** (Informal versions of Theorem 12 and Theorem 16). *For the quantum non-identical mean estimation problem with sufficiently small accuracy ϵ,*

- *if all random variables are bounded in $[L, H]$ and $m = \Omega(\log(\frac{H-L}{\epsilon}))$, there is a quantum algorithm that estimates μ to within additive error ϵ if $T = \Omega(\frac{H-L}{\epsilon})$. The algorithm uses $O(\frac{H-L}{\epsilon}\log(\frac{H-L}{\epsilon}))$ samples in total;*
- *if all random variables are sub-Gaussian with parameter K and $m = \tilde{\Omega}(\log(\frac{K}{\epsilon}))$, there is a quantum algorithm that estimates μ to within additive error ϵ if $T = \tilde{\Omega}(\frac{K}{\epsilon})$. The algorithm uses $\tilde{O}(\frac{K}{\epsilon})$ samples in total.*

In the worst case, the variance of random variables bounded in $[L, H]$ can be $(H - L)^2/4$, so the optimal classical estimator needs $\Theta((H - L)^2/\epsilon^2)$ samples to estimate μ up to additive error ϵ. For normal random variables, their sub-Gaussian parameter K equals their standard deviation σ, so the optimal classical estimator needs $\Theta(K^2/\epsilon^2)$ samples to estimate μ up to additive error ϵ. Therefore, the quantum estimators in Theorem 1 achieve nearly quadratic speed-up compared to classical estimators.

On the other hand, for $m = 1$, we show that any algorithm with relatively small working register have no speed-up compared to classical estimators.

▶ **Theorem 2** (Informal version of Theorem 23). *Suppose all random variables in the quantum non-identical mean estimation problem with $m = 1$ have mean bounded by R and variance bounded by σ^2. Let $\mathcal{A}$ be a quantum query algorithm acting on query register Q, working register W such that the number of qubits in Q is larger than that in W by $\Omega(\log(\frac{R}{\epsilon}))$. It requires $T = \Omega(\frac{\sigma^2}{\epsilon^2})$ if there exists an algorithm $\mathcal{A}$ solving this problem. The sample complexity of $\mathcal{A}$ is $T = \Omega(\frac{\sigma^2}{\epsilon^2})$.*

For general $m \geq 1$, we give another sample complexity lower bound of estimating mean of Bernoulli random variables.

▶ **Theorem 3** (Informal version of Theorem 25). *Suppose all random variables in the quantum non-identical mean estimation problem with $m \geq 1$ are Bernoulli random variables with mean $\mu \in (0, 1)$, and the accuracy ϵ satisfies $\epsilon \leq \mu(1 - \mu)$ and $\epsilon = O(\frac{1}{m^2})$. It requires $T = \Omega(\frac{1}{\epsilon m^2})$ if there exists a quantum query algorithm solving this problem. The sample complexity is $mT = \Omega(\frac{1}{\epsilon m})$ in total.*

In Theorem 3, we take the Bernoulli random variables as a hard instance for the quantum non-identical mean estimation problem. Note that if $\epsilon = \Theta(\mu(1 - \mu))$, the classical optimal estimator needs $\Theta(\frac{\mu(1-\mu)}{\epsilon^2}) = \Theta(\frac{1}{\epsilon})$ samples to estimate the mean of the Bernoulli random variable. Therefore, Theorem 3 shows that there is no quantum speed-up in this case if $m = O(1)$. However, it does not rule out the possibility of quantum speed-up for estimating the mean of Bernoulli random variables with $\epsilon = o(\mu(1-\mu))$ or $m = \Omega(1)$. For example, if $\mu = \Theta(1), \epsilon = o(1)$, and $m = \Omega(\log(\frac{1}{\epsilon}))$, the quantum estimator for bounded random variables in Theorem 1 can estimate μ up to error ϵ using $O(\frac{1}{\epsilon}\log(\frac{1}{\epsilon}))$ samples while classical estimators need $\Omega(\frac{1}{\epsilon^2})$ samples.

In addition, Theorem 2 and Theorem 3 give two different lower bounds when $m = 1$. Compared with Theorem 3, the lower bound in Theorem 2 matches the classical upper bound for general distributions with variance σ^2, but an additional requirement is that the register W has relatively small dimension.

Finally, we use Bernoulli random variable as an example to summary our systematical investigation on the quantum non-identical mean estimation problem.

▶ **Corollary 4.** *For Bernoulli random variable with mean μ such that $\epsilon = \Theta(\mu(1-\mu))$,*

- *if $m = \Omega(\log(1/\epsilon))$ and $T = \Omega(1/\epsilon)$, there exists an algorithm solving this problem using $O(\frac{1}{\epsilon}\log(1/\epsilon))$ quantum samples, achieving a near-quadratic speed-up;*
- *if $m = \Omega(\log(1/\epsilon))$ and $T = o(1/\epsilon m^2)$, there is no quantum algorithm solving this problem. There is an additional requirement that $\epsilon = O(1/m^2)$;*
- *if $m = O(1)$, there is no quantum speed-up for this problem.*

Proof. This corollary comes directly from Theorem 1, Theorem 2, and Theorem 3. ◀

1.2 Techniques

1.2.1 Upper Bound

From a high-level perspective, our quantum algorithms for non-identical mean estimation encode the mean to an amplitude, use an uncomputation trick to be introduced below to align different oracles, and then use amplitude estimation to estimate the mean.

We start with the bounded case. Recall that this paper studies non-identically distributed samples and assumes that we have access to unitaries $O_{X_1}, \ldots, O_{X_T}$, where

$$O_{X_i}|\mathbf{0}\rangle = \sum_{x\in E_i} \sqrt{p_i(x)}|\psi_x^{(i)}\rangle|x\rangle. \tag{4}$$

The mean $\mu = \mu_i = \sum_{x\in E_i} p_i(x)x$ is equal for different $i \in [T]$ (In fact, these μ_i can be slightly different – see Remark 13 for more details), but each O_{X_i} has potentially different garbage states $|\psi_x^{(i)}\rangle$ and each can only be used for very limited times. Suppose that for any $i \in [T]$, the bounded random variable X_i satisfies $X_i \in [L, H]$. If we have sufficient access to any specific O_{X_i}, we can construct a unitary

$$U_i|\mathbf{0}\rangle|0\rangle = \sqrt{q}|\psi_1^{(i)}\rangle|1\rangle + \sqrt{1-q}|\psi_0^{(i)}\rangle|0\rangle \tag{5}$$

by one call to O_{X_i} and a series of controlled rotations [17], where $q = (\mu - L)/(H - L)$. Consequently, the mean is encoded to an amplitude and direct amplitude estimation provides mean estimation with quadratic quantum speedup. However, in the non-identical case, we do not have sufficient number of calls to any specific U_i to provide quadratic speedup. Furthermore, it is very difficult to use a mixture of different U_i in amplitude estimation [3]. This is due to the reason that amplitude estimation is based on Grover's algorithm [9], which is essentially rotation in a two-dimensional plane spanned by two specific quantum states related to U_i. In our case, different U_i may have different $|\phi_1^{(i)}\rangle$ and $|\phi_0^{(i)}\rangle$, which forms different rotation planes and thus their mixed use is invalid. However, we can use a small number of calls to U_i to construct a unitary such that

$$S_i|\mathbf{0}\rangle = \sqrt{1-\epsilon_i}|0\rangle\left(\sqrt{r}|1\rangle + \sqrt{1-r}|0\rangle\right) + \sqrt{\epsilon_i}|1\rangle|\mathrm{garbage}_i\rangle \tag{6}$$

with r being a bijective function of q (the concrete value to be shown later) and ϵ_i being sufficiently small. Since the garbage state is small enough to be handled as an approximation error, S_i can be seen as an approximation of an unitary $S\colon |0\rangle \to \sqrt{r}|1\rangle + \sqrt{1-r}|0\rangle$. Therefore, We can then use these S_i instead of S to perform amplitude estimation, which provides estimation for r and thus q and μ.

The construction of S_i can be accomplished by an uncomputation trick [6] and fixed-point search [22]. Specifically, the uncomputation trick is to perform a unitary

$$V_i = (U_i^\dagger \otimes I)(I \otimes \mathrm{CNOT})(U_i \otimes I) \tag{7}$$

instead of U_i, which enjoys a property that it extracts the value of q separated from a garbage state related to $|\phi_1^{(i)}\rangle$ and $|\phi_0^{(i)}\rangle$. The computing result of $\langle b|\langle 0|\langle \mathbf{0}|V_i|\mathbf{0}\rangle|0\rangle|0\rangle$ for $b \in \{0,1\}$ tells that $V_i|\mathbf{0}\rangle|0\rangle|0\rangle$ only has components $|\mathbf{0}\rangle|0\rangle|0\rangle$, $|\mathbf{0}\rangle|0\rangle|1\rangle$, and a garbage state orthogonal to them. Besides, the amplitudes of the first two components are determined by q. In particular, it satisfies

$$\begin{aligned} V_i|\mathbf{0}\rangle|0\rangle|0\rangle = \sqrt{2q^2-2q+1}|\mathbf{0}\rangle|0\rangle\left(\frac{q}{\sqrt{2q^2-2q+1}}|1\rangle + \frac{1-q}{\sqrt{2q^2-2q+1}}|0\rangle\right) \\ + \sqrt{2q-2q^2}|\mathrm{garbage}_i\rangle, \end{aligned} \tag{8}$$

where $|\mathrm{garbage}_i\rangle$ is a unit garbage state and $(I \otimes \langle 0|\langle \mathbf{0}|)|\mathrm{garbage}_i\rangle = 0$. Therefore, we can use fixed-point quantum search [22] to stably amplify the amplitude of the state $\frac{q}{\sqrt{2q^2-2q+1}}|1\rangle + \frac{1-q}{\sqrt{2q^2-2q+1}}|0\rangle$ and thus S_i is constructed with $r = \frac{q^2}{2q^2-2q+1}$. See Theorem 12 for more details.

For a sub-Gaussian random variable with the absolute value of mean bounded by the sub-Gaussian parameter K, the probability of the random variable being more than a threshold related to K is sufficiently small and the mean of a truncated random variable can be a good enough approximation. Therefore, this case can be reduced to the case of bounded random variables. For general sub-Gaussian random variables $X_1, \ldots, X_T$, a constant number of classical experiments provide an estimation $\hat{\mu}$ within K-additive error, thus $X_1 - \hat{\mu}, \ldots, X_T - \hat{\mu}$ are sub-Gaussian random variables with the absolute value of mean bounded by K, which has been solved (see Theorem 16 for more details).

1.2.2 Lower Bound

We prove our two quantum query lower bounds using different techniques: the case $m = 1$ (Theorem 2) is established by simulating non-identical oracles by parallel oracles, and the case $m \geq 1$ (Theorem 3) is established by an adversarial method with non-identical oracles.

Simulating T Non-Identical Oracles by Constant T-Parallel Oracles

For the quantum non-identical mean estimation problem with $m = 1$, we give a sample complexity lower bound in Theorem 23 by constructing a quantum circuit with constant query depth simulating the original quantum circuit querying non-identical oracles. For any quantum query algorithm $\mathcal{A}$ using the *state preparation oracle* U_x such that the state

$U_x|\mathbf{0}\rangle$ encodes the input, suppose that there is a sequence of unitary oracles that maps $|\mathbf{0}\rangle$ to the same state but have different effects acting on other states orthogonal to $|\mathbf{0}\rangle$. Suppose that the working register of $\mathcal{A}$ is relatively small and $\mathcal{A}$ queries T non-identical oracles. In Theorem 21, we prove that for any projection Π with small image space, there is a quantum algorithm $\mathcal{A}'$ using two T-parallel queries such that

$$\|\Pi\mathcal{A}|\mathbf{0}\rangle\|^2 = \|(\Pi \otimes \langle\mathbf{0}|)\mathcal{A}'|\mathbf{0}\rangle|\mathbf{0}\rangle\|^2, \tag{9}$$

where a T-parallel query is to query T oracles simultaneously. This theorem builds a bridge between quantum algorithms with non-identical state preparation oracles and quantum algorithms with low query depth. If for any input x correct outputs of $\mathcal{A}$ lie in a small space V_x, and let $\mathrm{Im}(\Pi) = V_x$, then Theorem 21 shows that $\mathcal{A}$ and $\mathcal{A}'$ have the same probability to output a correct answer.

In Theorem 23, we prove that any quantum query algorithm $\mathcal{A}$ starting from an efficiently preparable state $|\mathbf{0}\rangle$ can be modified to recover the query register to $|\mathbf{0}\rangle$ with a small overhead. This reduces the dimension of the subspace that the correct outputs of $\mathcal{A}$ lie in, and then we use Theorem 21 to give a sample complexity lower bound of the quantum non-identical mean estimation problem with $m = 1$ based on the facts that parallelization only brings classical advantage to solving the quantum approximate counting problem [4], and the quantum approximate counting problem can be reduced to estimating the mean of Bernoulli random variables.

Adversarial Method with Non-Identical Oracles

Given a boolean function $f\colon \{0,1\}^n \to \{0,1\}$ and access to a unitary oracle O_x which encodes the information of some $x \in \{0,1\}^n$, the *generalized adversarial method* [13] gives a tight query complexity lower bound of computing $f(x)$. For any quantum query algorithm $\mathcal{A}$ and $x \in \{0,1\}^n$, let $|\psi_x^{(t)}\rangle$ be the quantum state after $\mathcal{A}$ queries O_x for t times. Suppose $\mathcal{A}$ can compute $f(x)$ with high probability for all $x \in \{0,1\}^n$ using T queries, then we have $\langle\psi_x^{(T)}|\psi_y^{(T)}\rangle = 1 - \Omega(1)$ for all $x \in f^{-1}(0)$ and $y \in f^{-1}(1)$. Since $\langle\psi_x^{(0)}|\psi_y^{(0)}\rangle = 1$, to give a lower bound of T, it suffices to give an upper bound on the *progress* at time t, $\langle\psi_x^{(t-1)}|\psi_y^{(t-1)}\rangle - \langle\psi_x^{(t)}|\psi_y^{(t)}\rangle$, for all $x \in f^{-1}(0)$, $y \in f^{-1}(1)$, and $t \in [T]$. The generalized adversarial method assigns a weight Γ_{xy} to every pair of $x \in f^{-1}(0)$, $y \in f^{-1}(1)$, which proves an upper bound for the weighted progress at time t:

$$S_{t-1} - S_t = \sum_{x\in f^{-1}(0),\ y\in f^{-1}(1)} \Gamma_{xy}(\langle\psi_x^{(t-1)}|\psi_y^{(t-1)}\rangle - \langle\psi_x^{(t)}|\psi_y^{(t)}\rangle), \tag{10}$$

and hence gives a lower bound on T. However, they regard $|\psi_x^{(t-1)}\rangle, |\psi_y^{(t-1)}\rangle$ as free variables independent of previous states $|\psi_x^{(t')}\rangle, |\psi_y^{(t')}\rangle$ for $t' < t - 1$ while bounding the weighted progress at t, so their upper bound of $S_{t-1} - S_t$ is independent of t. Therefore, if the algorithm queries different oracles at different times, the adversarial method cannot give better lower bound than the case that all oracles are the same. In Lemma 24, we apply the adversarial method on the quantum approximate counting problem, but analyze the progress in another way which utilizes the connection between $|\psi_x^{(t)}\rangle$ and $|\psi_x^{(t')}\rangle$ for different t and t'. Specifically, we show that any quantum query algorithm solving the quantum approximate counting problem has progress upper bounded by $O(\frac{t}{n})$ at time t, where n is the number of items. The original adversarial method gives an $O(\frac{1}{\sqrt{n}})$ upper bound of the progress at any time t. Boyer et al. [2] gave a similar analysis of quantum search which utilizes the connection between states at different time t, and got a tight lower bound of quantum search with a better constant factor compared to the hybrid argument. Since Reichardt [19] proved that the generalized adversarial method is asymptotically tight, we cannot expect more by exploring connections between states at different time with identical query oracles. However, if each oracle can only be queried a limited number of times, our bound in Lemma 24 is better than that obtained by the generalized adversarial method, since the progress bound $O(\frac{t}{n})$ is smaller in the early stages of the algorithm. We use this result to prove a query complexity lower bound of the quantum approximate counting problem with non-identical oracles. Since the quantum approximate counting problem can be reduced to estimating the mean of a Bernoulli random variable, we get a sample complexity lower bound of the quantum non-identical mean estimation problem in Theorem 3 for general m.

1.3 Organization

The rest of the paper is organized as follows. In Section 2 we formally define the input model and the quantum non-identical mean estimation problem, introduce the concept of parallel quantum query algorithms, and introduce quantum subroutines used in our algorithms. In Section 3 we give quantum algorithms for estimating the mean of non-identically distributed bounded or sub-Gaussian random variables with quadratic speed-up. In Section 4 we give two quantum query lower bounds of the quantum non-identical mean estimation problem based on reductions to low-depth quantum algorithms and the adversarial method with non-identical oracles, respectively.

2 Preliminaries

2.1 Notations

We denote $\{1,2,\ldots,n\}$ by $[n]$. We use $|\psi\rangle_{A,B}$ to indicate that the state $|\psi\rangle$ is in quantum registers A and B. For a quantum register A, we denote its number of qubits by n_A. For a boolean string $x \in \{0,1\}^n$, we denote its Hamming weight $|\{i \in [n] \mid x_i = 1\}|$ by $|x|$. We abbreviate $|0^k\rangle$ as $|\mathbf{0}\rangle$ if k can be inferred from the context.

2.2 Input Model

We first recall the definition of random variables and the input model of the classical mean estimation problem.

▶ **Definition 5** (Random variable). *A finite random variable X is a function $X\colon \Omega \to E$ for some probability space (Ω, p), where Ω is the finite sample space, p is a probability measure on Ω, and $E \subset \mathbb{R}$.*

Next, we assume that the random variable is the output of a quantum process O_X, and we can query O_X as an oracle to access X.

▶ **Definition 6** (Quantum random variable). *For any finite random variable X, a quantum random variable encoding X is a pair $(\mathcal{H}, O_X)$, where $\mathcal{H}$ is a Hilbert space and O_X is a unitary operator on $\mathcal{H}$ that performs the mapping*

$$O_X|\mathbf{0}\rangle = \sum_{x \in E} \sqrt{p(x)}|\psi_x\rangle|x\rangle \tag{11}$$

for some unknown garbage unit state $|\psi_x\rangle$.

Following the notation in [11], we call each application to U and $U^\dagger$ a *quantum experiment*. We use the number of quantum experiments to measure the sample complexity of a quantum query algorithm.

▶ **Definition 7** (Quantum experiment). *Let $(\mathcal{H}, O_X)$ be a quantum random variable. A quantum experiment is the process of applying O_X or its inverse $O_X^\dagger$ or their controlled versions to a state in $\mathcal{H}$.*

Performing a quantum experiment of a quantum random variable $(\mathcal{H}, O_X)$ can be regarded as a query to the unitary oracle O_X in the quantum query model, so the sample complexity is equivalent to the query complexity in this context, and we use the two terms interchangeably.

This input model is widely used in previous quantum mean estimation algorithms. The same oracle as defined in Definition 6 is used in [17]. Kothari and O'Donnell [16] used a similar input model except that they encode the probability distribution and the random variable mapping $\Omega \to \mathbb{R}$ in two oracles separately, and their algorithm also works well with the oracle in Definition 6. Hamoudi and Magniez [12, 11] used a more general input model called "q-random-variable", where the value of the random variable is implicitly encoded in a register and can be compared with a constant or performed conditional Pauli rotations, and our oracle can be regarded as an instance of the "q-random-variable". Since the oracle in Definition 6 already covers many common cases, we use it instead of the "q-random-variable" for simplicity and clarity. In fact, our quantum algorithm in Theorem 12 can also apply to the general "q-random-variable".

The unitary O_X is a quantum generalization of the process generating a sample of X. Bennett [1] proved that any classical algorithm using time T and space S can be modified to be a reversible algorithm using time $O(T)$ and space $O(ST^\epsilon)$ for any $\epsilon > 0$, and hence can be simulated by a quantum circuit. Therefore, for any randomized algorithm $\mathcal{A}$, we can implement the oracle O_X in Definition 6 encoding the output distribution of $\mathcal{A}$ with a small overhead.

Another natural way for a quantum algorithm to access a random variable is to assume that several copies of $|\psi_X\rangle = \sum_{x\in E} \sqrt{p(x)}|x\rangle$ encoding the information of X are given as the initial quantum state. This model is weaker than the one in Definition 6 since it does not provide access to a unitary preparing $|\psi_X\rangle$. Hamoudi [11] demonstrated that there is no quantum speed-up for the original mean estimation problem in this model. Therefore, it can be inferred that there is no quantum speed-up for the mean estimation problem of non-identically distributed random variables in this model, as it is a harder problem.

Based on the definition of quantum random variable, we define the mean estimation problem of non-identically distributed random variables formally as the following task.

▶ **Task 8** (Quantum non-identical mean estimation). *Let $(\mathcal{H}, O_{X_1}), \ldots, (\mathcal{H}, O_{X_T})$ be a sequence of quantum random variables on the same Hilbert space $\mathcal{H}$. Assume there exists μ and $\delta \in (0,1)$ such that each $\mu_i := \mathbb{E}[X_i]$ satisfies $|\mu_i - \mu| \le \delta$ for all $i \in [T]$. Given the repetition parameter $m \in \mathbb{N}$ and accuracy ϵ such that $\delta < c\epsilon$ for some constant $c < 1$, the quantum non-identical mean estimation problem is to estimate μ to within additive error ϵ with probability at least $2/3$ using each O_{X_i} or $O_{X_i}^\dagger$ or their controlled versions at most m times.*

The non-identity of quantum random variables means more than the non-identity of classical random variables. Specifically, the difference between two quantum random variables $(\mathcal{H}, O_X), (\mathcal{H}, O_Y)$ lies in the following three aspects: the results of applying O_X and O_Y to states orthogonal to $|\mathbf{0}\rangle$, the garbage state $|\psi_x\rangle$, and the random variables they encode. In contrast, the difference between two classical random variables is solely determined by the third aspect. Consequently, the quantum mean estimation problem of non-identically distributed random variables is more challenging than its classical counterpart.

2.3 Parallel Quantum Query Algorithms

The classical parallel algorithm implies that the algorithm can perform multiple operations simultaneously, which has become increasingly important in recent years with the development of multi-core processors. In the quantum setting, there is an additional reason to consider parallel algorithms: quantum states are fragile and susceptible to disruption by environmental factors, specifically decoherence. By reducing the computation time, parallel quantum algorithms can reduce the probability of decoherence. One example is parallel quantum query algorithms which can make multiple queries simultaneously, where a p-parallel query is defined as making p parallel queries simultaneously. Zalka [23] gave an algorithm that makes $\sqrt{\frac{n}{p}}$ p-parallel queries to solve the unstructured search problem with 1 marked item among n items and showed that its query complexity is optimal. Subsequent works also analyzed the parallel quantum query complexity of quantum search [10], quantum walk [15], quantum counting [4], and Hamiltonian simulation [24].

2.4 Quantum Subroutines

▶ **Lemma 9** (Approximating unitary operators, Eq. (4.63) of [18]). *Let $||\cdot||$ be the operator 2-norm. For unitary operators $\{U_i\}_{i=1}^m$, $\{V_i\}_{i=1}^m$, it holds that*

$$||U_mU_{m-1}\dots U_1 - V_mV_{m-1}\dots V_1|| \le \sum_{j=1}^{m} ||U_j - V_j||.$$

▶ **Lemma 10** (Amplitude estimation, Theorem 12 of [3]). *Given a unitary U satisfying*

$$U|\mathbf{0}\rangle = \sqrt{p}|\phi_1\rangle|1\rangle + \sqrt{1-p}|\phi_0\rangle|0\rangle \tag{12}$$

for some $p \in [0,1]$, there exists a quantum circuit C on a larger space such that the measurement outcome of $C|\mathbf{0}\rangle|\mathbf{0}\rangle$, $\tilde{p}$, satisfies

$$|\tilde{p} - p| \le \frac{2\pi\sqrt{p(1-p)}}{M} + \frac{\pi^2}{M^2} \tag{13}$$

with probability $\frac{8}{\pi^2}$, where C has M calls to the controlled versions of $I - 2U|\mathbf{0}\rangle\langle\mathbf{0}|U^\dagger$. Denote the algorithm by $\mathrm{AmpEst}(U, M)$.

▶ **Lemma 11** (Fixed-point quantum search, [22]). *Let A be a unitary and Π be an orthogonal projector such that $\Pi A|0\rangle = \lambda|\phi\rangle$, where $\lambda \in \mathbb{R}$ and $|\phi\rangle$ is a normalized quantum state. There exists a quantum circuit $S_L = \mathrm{FixSearch}(A, \Pi, \epsilon)$ such that $|||\phi\rangle - S_L|0\rangle|| \le \epsilon$, consisting of $O(\log(1/\epsilon)/\lambda)$ queries to A, $A^\dagger$, and $\mathrm{C}_\Pi\mathrm{NOT}$. Here $\mathrm{C}_\Pi\mathrm{NOT}$ is the Π-controlled* NOT *operator*

$$\mathrm{C}_\Pi\mathrm{NOT} = X \otimes \Pi + I \otimes (I - \Pi),$$

where X is the Pauli-X matrix.

3 Upper Bound

In this section, we first introduce an algorithm that solves Task 8 for bounded random variables, and then generalize it to sub-Gaussian variables.

Algorithm 1 Mean Estimation of Bounded Random Variables.

```
1: Input: sequence of random variable oracle $\{O_{X_i}\}_{i=1}^{T}$, accuracy $\epsilon$, mean difference $\delta$,
   repetition parameter $m$, lower bound $L$, upper bound $H$
2: Output: mean estimation $\tilde{\mu}$
   // Construct quantum circuit $S_i$
3: Construct unitary $U_i$
   $$U_i : |\mathbf{0}\rangle|0\rangle \xrightarrow{O_{X_i}\otimes I} \sum_{x\in E_i} \sqrt{p_i(x)}|\psi_x^{(i)}\rangle|x\rangle|0\rangle$$
   $$\xrightarrow{\text{controlled rotation}} \sum_{x\in E_i} \sqrt{p_i(x)}|\psi_x^{(i)}\rangle|x\rangle \left(\sqrt{\frac{x-L}{H-L}}|1\rangle + \sqrt{\frac{H-x}{H-L}}|0\rangle\right)$$
4: Let $V_i = (U_i^\dagger \otimes I)(I \otimes \mathrm{CNOT})(U_i \otimes I)$
5: Let $S_i = \mathrm{FixSearch}(V_i, |\mathbf{0}\rangle|0\rangle\langle 0|\langle\mathbf{0}| \otimes I, \epsilon' = O(\epsilon^2/(H-L)^2))$
   // Mean estimation using $S_i$
6: Let $\tilde{p}$ be the output of $\mathrm{AmpEst}(S, M = O(\frac{H-L}{\epsilon}))$, where $S$ is arbitrarily replaced by
   $S_1, \ldots, S_T$.
7: Output $\tilde{\mu} = \frac{\tilde{p}-\sqrt{\tilde{p}(1-\tilde{p})}}{2\tilde{p}-1}(H-L) + L$
```

3.1 Mean Estimation of Bounded Random Variables

In this subsection, we introduce an algorithm that solves Task 8 with quadratic speed-up given the condition that random variables $X_1, \ldots, X_T$ are bounded in $[L, H]$. According to the task, for each $i \in [T]$, oracle O_{X_i} can be used at most m times.

For clarity, we describe the algorithm with two phases. Let

$$|\phi_i\rangle = \frac{q_i}{\sqrt{2q_i^2 - 2q_i + 1}}|1\rangle + \frac{1-q_i}{\sqrt{2q_i^2 - 2q_i + 1}}|0\rangle.$$

Here $q_i = \frac{\mu_i - L}{H-L} \in [0, 1]$. For each $i \in [T]$, We will construct a quantum circuit S_i that satisfies $S_i|\mathbf{0}\rangle \approx |\phi_i\rangle$ with m calls to O_{X_i}. Then we will prove that performing amplitude estimation with these S_i gives an ϵ-additive estimation of μ.

▶ **Theorem 12.** *Assume that all random variables $X_1, \ldots, X_T$ in Task 8 are bounded in $[L, H]$. Let m, ϵ, δ in Algorithm 1 satisfy $m = \Omega(\log(\frac{H-L}{\epsilon}))$, $\epsilon = O(\frac{(\mu-L)(H-\mu)}{H-L})$, and $\delta < \epsilon/2$. Algorithm 1 solves this task if $T = \Omega(\frac{H-L}{\epsilon})$, using $O(\frac{H-L}{\epsilon}\log(\frac{H-L}{\epsilon}))$ quantum experiments in total.*

Proof. We first prove that S_i in Line 5 satisfies $S_i|\mathbf{0}\rangle|0\rangle|0\rangle = \sqrt{1-\epsilon_i}|\mathbf{0}\rangle|0\rangle|\phi_i\rangle + \sqrt{\epsilon_i}|\mathrm{garbage}_i\rangle$. According to the construction of U_i in Line 3 of Algorithm 1, we have

$$U_i|\mathbf{0}\rangle|0\rangle = \sqrt{q_i}|\psi_1^{(i)}\rangle|1\rangle + \sqrt{1-q_i}|\psi_0^{(i)}\rangle|0\rangle \tag{14}$$

for some unit states $|\psi_1^{(i)}\rangle$ and $|\psi_0^{(i)}\rangle$. Consider the V_i in Line 4 where we append a qubit to the register. For any $b \in \{0, 1\}$ we have

$$\begin{aligned}\langle b|\langle 0|\langle \mathbf{0}|V_i|\mathbf{0}\rangle|0\rangle|0\rangle &= ((U_i\otimes I)|\mathbf{0}\rangle|0\rangle|b\rangle)^\dagger (I\otimes \mathrm{CNOT})(U_i\otimes I)|\mathbf{0}\rangle|0\rangle|0\rangle \\ &= \left(\sqrt{q_i}\langle b|\langle 1|\langle\psi_1^{(i)}| + \sqrt{1-q_i}\langle b|\langle 0|\langle\psi_0^{(i)}|\right)\left(\sqrt{q_i}|\psi_1^{(i)}\rangle|1\rangle|1\rangle + \sqrt{1-q_i}|\psi_0^{(i)}\rangle|0\rangle|0\rangle\right) \\ &= \begin{cases} q_i & b=1 \\ 1-q_i & b=0, \end{cases} \end{aligned} \tag{15}$$

which implies that

$$\begin{aligned} V_i|\mathbf{0}\rangle|0\rangle|0\rangle = \sqrt{2q_i^2-2q_i+1}|\mathbf{0}\rangle|0\rangle\left(\frac{q_i}{\sqrt{2q_i^2-2q_i+1}}|1\rangle + \frac{1-q_i}{\sqrt{2q_i^2-2q_i+1}}|0\rangle\right) \\ + \sqrt{2q_i-2q_i^2}|\mathrm{garbage}_i\rangle, \end{aligned} \tag{16}$$

where $|\mathrm{garbage}_i\rangle$ is a unit garbage state and $(I\otimes\langle 0|\langle\mathbf{0}|)|\mathrm{garbage}_i\rangle = 0$. Moreover, we define

$$|\phi_i\rangle = \frac{q_i}{\sqrt{2q_i^2-2q_i+1}}|1\rangle + \frac{1-q_i}{\sqrt{2q_i^2-2q_i+1}}|0\rangle, \qquad |s_i\rangle = V_i|\mathbf{0}\rangle|0\rangle|0\rangle. \tag{17}$$

Under these notations, we have

$$(|\mathbf{0}\rangle|0\rangle\langle 0|\langle\mathbf{0}|\otimes I)\, V_i|\mathbf{0}\rangle|0\rangle|0\rangle = \sqrt{2q_i^2-2q_i+1}|\mathbf{0}\rangle|0\rangle|\phi_i\rangle. \tag{18}$$

Together with Lemma 11 and the fact that $\sqrt{2q_i^2-2q_i+1}\geq\frac{1}{\sqrt{2}}$, we know that S_i in Line 5 satisfies

$$S_i|\mathbf{0}\rangle|0\rangle|0\rangle = \sqrt{1-\epsilon_i}|\mathbf{0}\rangle|0\rangle|\phi_i\rangle + \sqrt{\epsilon_i}|\mathrm{garbage}_i\rangle, \tag{19}$$

where $\epsilon_i\leq\epsilon'$ and S_i contains $O\big(\log\frac{1}{\epsilon'}\big) = O\big(\log\big(\frac{H-L}{\epsilon}\big)\big)$ calls to V_i.

Let

$$q = \frac{\mu-L}{H-L}\in[0,1], \qquad |\phi\rangle = \frac{q}{\sqrt{2q^2-2q+1}}|1\rangle + \frac{1-q}{\sqrt{2q^2-2q+1}}|0\rangle,$$

and S be a unitary such that

$$S|\mathbf{0}\rangle|0\rangle|0\rangle = |\mathbf{0}\rangle|0\rangle|\phi\rangle. \tag{20}$$

Performing an amplitude estimation using $\{S_i\}_{i=1}^T$ provides a result similar to an amplitude estimation using S, and thus provides a mean estimation with additive error $O(\epsilon)$. See the details in [14] appendix A.1.

Each V_i uses two quantum experiments, each S_i uses $O(\log\big(\frac{H-L}{\epsilon}\big))$ calls to V_i, and C' uses $M = O(\frac{H-L}{\epsilon})$ calls to controlled S_i. Therefore, the total number of quantum experiments is $O\big(\frac{H-L}{\epsilon}\log\big(\frac{H-L}{\epsilon}\big)\big)$. ◀

▶ Remark 13. For every $i\in[T]$, S_i can be seen as an approximation of unitary S. The slight difference δ among different μ_i only causes a part of approximation error which is bounded by ϵ. Therefore, this difference is tolerable in our algorithm. See [14] equation (73) and (78) for more details.

3.2 Mean Estimation of Sub-Gaussian Random Variables

In this subsection, we consider the quantum non-identical mean estimation problem of sub-Gaussian random variables.

Algorithm 2 Mean Estimation of Mean-Bounded sub-Gaussian Random Variable.

```
Input: sequence of random variable oracle $\{O_{X_i}\}_{i=1}^{T}$, accuracy $\epsilon$, mean difference $\delta$, repetition parameter $m$, upper bound for mean $R$, sub-Gaussian parameter $K$
Output: mean estimation $\tilde{\mu}$
Let $\Delta = K \max\left\{\sqrt{4\log\left(\frac{128K}{\epsilon}\right)}, \sqrt{2\log\left(\frac{32R}{\epsilon}\right)}\right\}$, $L = -R - \Delta$, $H = R + \Delta$
Construct unitary $O_{\tilde{X}_i}$
    $$O_{\tilde{X}_i} : |\mathbf{0}\rangle|\mathbf{0}\rangle \xrightarrow{O_{X_i}\otimes I} \sum_{x\in E_i} \sqrt{p_i(x)}|\psi_x^{(i)}\rangle|x\rangle|\mathbf{0}\rangle$$
    $$\xrightarrow{\text{CNOT}} \sum_{x\in[L,H]} \sqrt{p_i(x)}|\psi_x^{(i)}|x\rangle|x\rangle + \sum_{x\in E_i\setminus[L,H]} \sqrt{p_i(x)}|\psi_x^{(i)}|x\rangle|\mathbf{0}\rangle$$
Output $\tilde{\mu}$ =Algorithm 1($\{O_{\tilde{X}_i}\}_{i=1}^{T}$, accuracy $\epsilon$, mean difference $\delta = \epsilon/2$, $m$, $L$, $H$)
```

▶ **Definition 14.** *A random variable X is sub-Gaussian with parameter K if for all $t \geq 0$*

$$\mathbb{P}[|X - \mathbb{E}[X]| \geq t] \leq 2\exp\left(-\frac{t^2}{2K^2}\right). \tag{21}$$

We first give a quantum algorithm estimating the mean of non-identically distributed sub-Gaussian random variables with quadratic speed-up if the mean of the random variables are bounded by their sub-Gaussian parameter. This case can be reduced to the case of bounded random variables by truncation. Then, we show that this algorithm can be generalized to any sub-Gaussian random variable.

▶ **Lemma 15.** *Suppose all random variables $X_1, \ldots, X_T$ in Task 8 are sub-Gaussian with parameter K and their mean satisfies $|\mu_i| \leq R$, $R \leq K$. Let m, R, K, ϵ, δ in Algorithm 2 satisfies that $m = \Omega\left(\log\left(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\right)\right)$, $\epsilon = O(K)$, and $\delta < \epsilon/4$. Algorithm 2 solves Task 8 if $T = \Omega(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon})$, using $O\left(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\log\left(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\right)\right)$ quantum experiments in total.*

Quantum random variable $\tilde{X}_i$ generated by oracle $O_{\tilde{X}_i}$ in Algorithm 2 is a truncated version of X_i. Calculation shows that the mean difference is within $\frac{\epsilon}{2}$, thus Algorithm 2 provides an estimation with $O(\epsilon)$ additive error.

Proof. See [14] appendix A.2. ◀

For general sub-Gaussian distributions, we first use $O(1)$ classical samples to estimate the mean of these sub-Gaussian random variables up to additive error $K/2$, and then shift the random variables by subtracting the approximate mean so that the shifted random variables have mean bounded by their sub-Gaussian parameter. After that, we can use Lemma 15 to estimate the mean of the shifted random variables.

▶ **Theorem 16.** *Assume all random variables $X_1, \ldots, X_T$ in Task 8 are sub-Gaussian with parameter K. Let m, K, δ, ϵ in Algorithm 3 satisfy that $m = \Omega\left(\log\left(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\right)\right)$, $\epsilon = O(K)$, and $\delta < \epsilon/4$. Algorithm 3 solves Task 8 if $T = \Omega(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon})$, using $O\left(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\log\left(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\right)\right)$ quantum experiments in total.*

Algorithm 3 Mean Estimation of sub-Gaussian Random Variable.

```
1: Input: sequence of random variable oracle $\{O_{X_i}\}_{i=1}^T$, accuracy $\epsilon$, repetition parameter $m$, sub-Gaussian parameter $K$
2: Output: mean estimation $\tilde{\mu}$
3: Perform $N = \lceil 8\log(20) \rceil$ times classical experiments on arbitrary $X_i$ and let the average of the samples be $\hat{\mu}$
4: Construct unitary $O_{X_i'}$

   $O_{X_i'} : |\mathbf{0}\rangle|\mathbf{0}\rangle \xrightarrow{O_{X_i}\otimes I} \sum_{x\in E_i} \sqrt{p_i(x)}|\psi_x^{(i)}|x\rangle|\mathbf{0}\rangle$
   $\longrightarrow \sum_{x\in E_i} \sqrt{p_i(x)}|\psi_x^{(i)}|x\rangle|x-\hat{\mu}\rangle$

5: Output $\tilde{\mu}$ =Algorithm 2($\{O_{X_i'}\}_{i=1}^T$, accuracy $\epsilon$, mean difference $\delta = \epsilon/4$, $m$, upper bound for mean $R = K$, sub-Gaussian parameter $K$)
```

Proof. Classical experiment in Line 3 can be naturally implemented by quantum access to random variable. For any $i \in [T]$, by applying O_{X_i} to $|\mathbf{0}\rangle$ and measuring the second register in computational basis, we can get a classical sample of X_i. Since $\hat{\mu}$ is the average value of $N = \lceil 8\log(20) \rceil$ samples, by the Hoeffding inequality for sub-Gaussian distributions [21], we have

$$\mathbb{P}[|\hat{\mu} - \mathbb{E}[\hat{\mu}]| \geq \frac{K}{2}] \leq 2\exp\Big(-\frac{N}{2K^2}\frac{K^2}{4}\Big) \leq \frac{1}{10}. \tag{22}$$

In addition, since $|\mu_i - \mu| \leq \delta$ for all $i \in [T]$, we have

$$|\mathbb{E}[\hat{\mu}] - \mu| \leq \delta. \tag{23}$$

O'_{X_i} can be seen as quantum query to random variable $X_i' = X_i - \hat{\mu}$. With probability at least $\frac{9}{10}$, we have

$$|\mathbb{E}[X_i']| = |\mathbb{E}[X_i] - \hat{\mu}| \leq |\mathbb{E}[X_i] - \mathbb{E}[\hat{\mu}]| + |\hat{\mu} - \mathbb{E}[\hat{\mu}]| \leq \delta + \frac{K}{2} \leq K. \tag{24}$$

Therefore, by Lemma 15 with $R = K$, $m = \Omega\Big(\log\Big(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\Big)\Big)$ and $X_i' = X_i - \hat{\mu}$, we can estimate $\mu - \hat{\mu}$ with additive error $O(\epsilon)$ with probability at least $\frac{4}{5}$ using $O\Big(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\log\Big(\frac{K\sqrt{\log(\frac{K}{\epsilon})}}{\epsilon}\Big)\Big)$ quantum experiments. Subtracting $\hat{\mu}$ from the estimate gives the final output of the algorithm which is an ϵ-additive estimate of μ with probability at least $\frac{4}{5}\cdot\frac{9}{10} \geq \frac{2}{3}$. ◀

4 Lower Bound

In this section, we prove sample complexity lower bounds for the quantum non-identical mean estimation problem in Task 8.

Let m be the repetition parameter defined Task 8. In Section 4.1, we give a sample complexity lower bound for $m = 1$, and show there is no quantum speed-up compared to classical algorithms. In Section 4.2, we give a sample complexity lower bound for $m \geq 1$.

4.1 Lower Bound for $m = 1$

Let X be a finite random variable with support E. Let $(\mathcal{H}, O_X)$ be a quantum random variable in Definition 6, i.e.,

$$O_X|\mathbf{0}\rangle = \sum_{x\in E}\sqrt{p(x)}|\psi_x\rangle|x\rangle, \tag{25}$$

and we denote the output state by $|\psi_X\rangle$. A p-parallel query to O_X is to apply the unitary $O_X^{\otimes q}$ or $O_X^{\dagger\otimes q}$ for $q \le p$.

Note that Eq. (25) only restricts the outcome of applying O_X on $|\mathbf{0}\rangle$, so the quantum random variable encoding the same X can be different. Throughout Section 4.1, we assume all quantum random variables encode the same finite random variable X. Given that $m = 1$, the algorithm can perform only one quantum experiment for each quantum random variable.

We use the quantum query model to analyze the sample complexity of the quantum non-identical mean estimation since every quantum experiment can be regarded as a query to the oracle O_X. A T-query quantum algorithm starts from an all-0 state $|\mathbf{0}\rangle_Q|\mathbf{0}\rangle_W$, and then interleaves fixed unitary operations $U_0, U_1, \dots, U_T$ with queries. Suppose different oracles are queried at different time, and we denote the t-th oracle queried by the algorithm as $O_X^{(t)}$. Without loss of generality, we assume that all queries are applied to register $|\mathbf{0}\rangle_Q$ and $U_0, U_1, \dots, U_T$ are applied to $|\mathbf{0}\rangle_Q|\mathbf{0}\rangle_W$. Whether to apply $O_X^{(t)}$ or $(O_X^{(t)})^\dagger$ needs to be determined in advance, and the choices can be represented by T boolean variables $a_1, \dots, a_T \in \{-1, 1\}$ such that

$$(O_X^{(t)})^{a_t} = \begin{cases} O_X^{(t)} & \text{if } a_t = 1, \\ (O_X^{(t)})^\dagger & \text{if } a_t = -1. \end{cases} \tag{26}$$

For any $1 \le t \le T$, let

$$|\psi^{(t)}\rangle := U_t(O_X^{(t)})^{a_t}\cdots(O_X^{(1)})^{a_1}U_0|\mathbf{0}\rangle_Q|\mathbf{0}\rangle_W. \tag{27}$$

Hence the final state of the algorithm is $|\psi^{(T)}\rangle$.

At the end of the algorithm, we will measure $|\psi^{(T)}\rangle$ and let the projection onto the correct outputs be Π_c, and the success probability of the algorithm is hence

$$\|\Pi_c|\psi^{(T)}\rangle\|^2. \tag{28}$$

4.1.1 Reduction to Low-depth Quantum Algorithms

For a quantum circuit with oracles, the query depth is the maximum number of queries on any path from an input qubit to an output qubit. In this section, we prove that the behavior of a quantum algorithm querying T non-identical oracles can be simulated by a low query depth quantum algorithm with the same number of queries. Actually, we will show that the behavior of the algorithm can be simulated by a quantum circuit using two T-parallel queries.

For any $1 \le t \le T$, let

$$|\phi^{(t)}_{\text{beg}}\rangle := \begin{cases} |\mathbf{0}\rangle & \text{if } a_t = 1, \\ |\psi_X\rangle & \text{if } a_t = -1, \end{cases} \tag{29}$$

$$|\phi^{(t)}_{\text{end}}\rangle := \begin{cases} |\psi_X\rangle & \text{if } a_t = 1, \\ |\mathbf{0}\rangle & \text{if } a_t = -1, \end{cases} \tag{30}$$

so that

$$(O_X^{(t)})^{a_t}|\phi_{\text{beg}}^{(t)}\rangle = |\phi_{\text{end}}^{(t)}\rangle. \tag{31}$$

This is the only subspace that $(O_X^{(t)})^{a_t}$'s behavior is fixed and defined by Eq. (25).

For any $1 \le t \le T$, let

$$\Pi_{\text{beg}}^{(t)} := |\phi_{\text{beg}}^{(t)}\rangle\langle\phi_{\text{beg}}^{(t)}| \otimes I, \tag{32}$$

and

$$|\psi_{\text{eff}}^{(t)}\rangle_{Q,W} := (O_X^{(t)})^{a_t}\Pi_{\text{beg}}^{(t)}U_{t-1}(O_X^{(t-1)})^{a_{t-1}}\Pi_{\text{beg}}^{(t-1)}\cdots U_1(O_X^{(1)})^{a_1}\Pi_{\text{beg}}^{(1)}U_0|\mathbf{0}\rangle_Q|\mathbf{0}\rangle_W \tag{33}$$

$$= (|\phi_{\text{end}}^{(t)}\rangle\langle\phi_{\text{beg}}^{(t)}| \otimes I)U_{t-1}\cdots U_1(|\phi_{\text{end}}^{(1)}\rangle\langle\phi_{\text{beg}}^{(1)}| \otimes I)U_0|\mathbf{0}\rangle_Q|\mathbf{0}\rangle_W. \tag{34}$$

These states are fixed no matter what the queries $O_X^{(t)}$ are, since all queries in Eq. (33) are applied to the subspace that its behavior is defined by Eq. (25).

We show in the following lemma that $|\psi_{\text{eff}}^{(t)}\rangle$ can be prepared by a quantum algorithm using two t-parallel queries after post-selection.

▶ **Lemma 17.** *Given a T-query quantum algorithm acting on registers Q and W, for any $0 \le t \le T$, $|\psi_{\text{eff}}^{(t)}\rangle$ defined in Eq. (33) can be prepared by another quantum circuit V_t^{low} using two t-parallel queries to any unitary oracle O_X satisfying Eq. (25) after post-selection, namely,*

$$\big(I_{W,Q_t} \otimes \langle\mathbf{0}|_{Q_0,\ldots,Q_{t-1}}\big)V_t^{\text{low}}|\mathbf{0}\rangle_{W,Q_0,\ldots,Q_t}, \tag{35}$$

where $Q_0,\ldots,Q_t$ are $t+1$ registers with n_Q qubits.

Proof. For all $1 \le t < T$, from the definition of $|\psi_{\text{eff}}^{(t)}\rangle$, it can be written as

$$|\psi_{\text{eff}}^{(t)}\rangle = |\phi_{\text{end}}^{(t)}\rangle|\phi_{\text{W}}^{(t)}\rangle \tag{36}$$

for some unnormalized state $|\phi_{\text{W}}^{(t)}\rangle$, then we have

$$|\phi_{\text{end}}^{(t+1)}\rangle|\phi_{\text{W}}^{(t+1)}\rangle = |\psi_{\text{eff}}^{(t+1)}\rangle = (|\phi_{\text{end}}^{(t+1)}\rangle\langle\phi_{\text{beg}}^{(t+1)}| \otimes I)U_t|\phi_{\text{end}}^{(t)}\rangle|\phi_{\text{W}}^{(t)}\rangle. \tag{37}$$

Apply $\langle\phi_{\text{end}}^{(t+1)}| \otimes I$ to both sides we have

$$|\phi_{\text{W}}^{(t+1)}\rangle = (\langle\phi_{\text{beg}}^{(t+1)}| \otimes I)U_t|\phi_{\text{end}}^{(t)}\rangle|\phi_{\text{W}}^{(t)}\rangle. \tag{38}$$

Define

$$|\psi_{\text{eff}}^{(0)}\rangle = |\mathbf{0}\rangle|\mathbf{0}\rangle, \quad |\phi_{\text{end}}^{(0)}\rangle = |\mathbf{0}\rangle, \quad |\phi_{\text{W}}^{(0)}\rangle = |\mathbf{0}\rangle, \tag{39}$$

so that Eq. (36) and Eq. (38) also hold for $t = 0$.

To construct the required circuit, We prove the following stronger statement.

▶ **Statement 18.** *Let O_X be any unitary satisfying Eq. (25), and $U_0^{\text{low}},\ldots,U_T^{\text{low}}$ be a sequence of quantum circuits satisfying $U_0^{\text{low}} = I$ and*

$$U_{t+1}^{\text{low}} = \begin{cases} ((U_t)_{Q_t,W} \otimes I)\cdot(U_t^{\text{low}} \otimes (O_X)_{Q_{t+1}}) & \textit{if } a_{t+1} = 1, \\ ((U_t)_{Q_t,W} \otimes I)\cdot(U_t^{\text{low}} \otimes I_{Q_{t+1}}) & \textit{if } a_{t+1} = -1, \end{cases} \tag{40}$$

for all $0 \le t < T$*. The quantum circuit* U_t^{low} *can prepare* $|\psi_{\mathrm{eff}}^{(t)}\rangle$ *after post-selection, namely,*

$$|psi_{\mathrm{eff}}^{(t)}\rangle = \Big(I_{W,Q_t} \bigotimes_{i=1}^{t} \langle \phi_{\mathrm{beg}}^{(i)}|_{Q_{i-1}}\Big) U_t^{\mathrm{low}} |\mathbf{0}\rangle_{W,Q_0,\dots,Q_t}, \tag{41}$$

for any $0 \le t \le T$*.*

Proof. See [14] appendix B.1. ◀

The number of queries in U_t^{low} is $|\{a_i = 1 \mid i \in [t]\}|$. Let

$$V_t^{\mathrm{low}} = \bigotimes_{1 \le i \le t, a_i = -1} (O_X^{\dagger})_{Q_i} U_t^{\mathrm{low}}, \tag{42}$$

then from Eq. (41) we have

$$\big(I_{W,Q_t} \otimes \langle \mathbf{0}|_{Q_0,\dots,Q_{t-1}}\big) V_t^{\mathrm{low}} |\mathbf{0}\rangle_{W,Q_0,\dots,Q_t}, \tag{43}$$

for all $0 \le t \le T$.

The number of queries in V_t^{low} is

$$|\{a_i = 1 \mid i \in [t]\}| + |\{a_i = -1 \mid i \in [t]\}| = t. \tag{44}$$

Conditioning on the state in registers $Q_0, \dots, Q_{t-1}$ to be $|\mathbf{0}\rangle$, V_t^{low} prepares $|\psi_{\mathrm{eff}}^{(t)}\rangle_{Q_t,W}$ and uses two t-parallel queries. ◀

Next, we demonstrate that $U_T|\psi_{\mathrm{eff}}^{(T)}\rangle$ is the only useful component in the final state $|\psi^{(T)}\rangle$, since other parts can be controlled by $O_X^{(t)}$ to make the result worse. Before that, we prove the following useful lemma.

▶ **Lemma 19.** *For any* T*-query quantum algorithm acting on registers* Q*,* W*, and any finite random variable* X *on* (Ω, p)*, if* $\dim \mathcal{H}_Q > 2 \dim \mathcal{H}_W$*, there exists a sequence of quantum random variables* $(\mathcal{H}_Q, O_X^{(1)}), \dots, (\mathcal{H}_Q, O_X^{(T-1)})$ *such that for any* $0 \le t < T$

$$|\psi^{(t)}\rangle = |\phi_{\mathrm{beg}}^{(t+1)}\rangle |\phi_W^{(t+1)}\rangle + |\psi_{\perp}^{(t)}\rangle, \tag{45}$$

for some unnormalized state $|\psi_{\perp}^{(t)}\rangle$ *orthogonal to* $|\phi_{\mathrm{beg}}^{(t+1)}\rangle \otimes \mathcal{H}_W$*.*

Proof. By induction. See the details in [14] appendix B.2. ◀

Now we prove that $U_T|\psi_{\mathrm{eff}}^{(T)}\rangle$ is the only useful component in the final state $|\psi^{(T)}\rangle$.

▶ **Lemma 20.** *Suppose that* X *is a finite random variable. For any* T*-query quantum algorithm acting on registers* Q*,* W*, and any projection* Π_c*, if* $\dim \mathcal{H}_Q > 2 \dim \mathcal{H}_W$ *and* $\dim \mathcal{H}_Q \ge 2 \dim \mathrm{Im}(\Pi_c)$*, then there exists a sequence of quantum random variables* $(\mathcal{H}_Q, O_X^{(1)}), \dots, (\mathcal{H}_Q, O_X^{(T)})$ *such that*

$$\|\Pi_c |\psi^{(T)}\rangle\|^2 = \|\Pi_c U_T |\psi_{\mathrm{eff}}^{(T)}\rangle\|^2. \tag{46}$$

Proof. Note that

$$\begin{aligned}
|\psi^{(T)}\rangle &= U_T (O_X^{(T)})^{a_T} |\psi^{T-1}\rangle & (47)\\
&= U_T (O_X^{(T)})^{a_T} (|\phi_{\mathrm{beg}}^{(T)}\rangle |\phi_W^{(T)}\rangle + |\psi_{\perp}^{(T-1)}\rangle) & (48)\\
&= U_T |\phi_{\mathrm{end}}^{(T)}\rangle |\phi_W^{(T)}\rangle + U_T (O_X^{(T)})^{a_T} |\psi_{\perp}^{(T-1)}\rangle & (49)\\
&= U_T |\psi_{\mathrm{eff}}^{(T)}\rangle + U_T (O_X^{(T)})^{a_T} |\psi_{\perp}^{(T-1)}\rangle. & (50)
\end{aligned}$$

To satisfy Eq. (46), we need to find a unitary operator $O_X^{(T)}$ such that

$$\Pi_c U_T (O_X^{(T)})^{a_T} |\psi_\perp^{(T-1)}\rangle = 0, \tag{51}$$

which means

$$(O_X^{(T)})^{a_T} |\psi_\perp^{(T-1)}\rangle \in (U_T^\dagger \mathrm{Im}(\Pi_c))^\perp. \tag{52}$$

By the same method in the proof of Lemma 19, we can construct oracle $O_X^{(T)}$. By similar argument to Lemma 19, we can prove that if

$$\dim \mathcal{H}_Q > \dim \mathcal{H}_W + \dim \mathrm{Im}(\Pi_c), \tag{53}$$

there exists $O_X^{(T)}$ such that Eq. (46) holds. By assumptions that $\dim \mathcal{H}_Q > 2 \dim \mathcal{H}_W$ and $\dim \mathcal{H}_Q \geq 2 \dim \mathrm{Im}(\Pi_c)$, we can conclude that Eq. (46) holds. ◀

In conclusion, there exists a sequence of quantum random variables such that the output of a T-query quantum algorithm can be simulated by a quantum algorithm using two T-parallel queries.

▶ **Theorem 21.** *For any T-query quantum algorithm $\mathcal{A}$ acting on registers Q, W, and any projection Π_c, suppose that $\dim \mathcal{H}_Q > 2 \dim \mathcal{H}_W$ and $\dim \mathcal{H}_Q \geq 2 \dim \mathrm{Im}(\Pi_c)$. Let $|\psi^{(T)}\rangle$ be the final state of the algorithm. There exists another quantum circuit U^{low} using two T-parallel queries such that for any finite random variable X, there is a sequence of quantum random variables $(\mathcal{H}_Q, O_X^{(1)}), \ldots, (\mathcal{H}_Q, O_X^{(T)})$ satisfying*

$$\|\Pi_c |\psi^{(T)}\rangle\|^2 = \left\| \left(\Pi_c \otimes \langle \mathbf{0}|_{Q_0,\ldots,Q_{T-1}}\right) U^{\mathrm{low}} |\mathbf{0}\rangle_{W,Q_0,\ldots,Q_T} \right\|^2, \tag{54}$$

where $Q_0, \ldots, Q_T$ are $T+1$ registers with n_Q qubits.

Proof. Let V_T^{low} be the low-depth quantum circuit defined in Lemma 17, and U_T be the unitary in algorithm $\mathcal{A}$ at time step T. By Lemma 17, the unitary $U^{\mathrm{low}} = ((U_T)_{Q_T,W} \otimes I) V_T^{\mathrm{low}}$ satisfies

$$\left(I \otimes \langle \mathbf{0}|_{Q_0,\ldots,Q_{T-1}}\right) U^{\mathrm{low}} |\mathbf{0}\rangle_{W,Q_0,\ldots,Q_T} = U_T |\psi_{\mathrm{eff}}^{(T)}\rangle_{Q_T,W}. \tag{55}$$

By Lemma 20, there exists a sequence of quantum random variables $(\mathcal{H}_Q, O_X^{(1)}), \ldots, (\mathcal{H}_Q, O_X^{(T)})$ such that

$$\|\Pi_c |\psi^{(T)}\rangle\|^2 = \|\Pi_c U_T |\psi_{\mathrm{eff}}^{(T)}\rangle\|^2 = \left\| \left(\Pi_c \otimes \langle \mathbf{0}|_{Q_0,\ldots,Q_{T-1}}\right) U^{\mathrm{low}} |\mathbf{0}\rangle_{W,Q_0,\ldots,Q_T} \right\|^2. \tag{56}$$

◀

4.1.2 Lower Bounds for Low-depth Quantum Mean Estimation Algorithms

Given an input $x = x_0 \ldots x_{n-1} \in \{0,1\}^n$, the quantum query to it is a unitary O_x such that

$$O_x |i\rangle |b\rangle = |i\rangle |b \oplus x_i\rangle \tag{57}$$

for all $i \in [n]$ and $b \in \{0,1\}$.

The *approximate counting* problem is that given O_x, output an estimate of $|x|$ up to error ϵ with high probability. From another perspective, we can think of $[n]$ as a sample space Ω with uniform distribution P, and $X\colon \Omega \to \{0,1\}$ is a Bernoulli random variable such that $X(i) = x_i$, and the mean of X is

$$p = \frac{|x|}{n}. \tag{58}$$

Note that

$$|0\rangle|0\rangle \xrightarrow{\text{Hardmard gates}} \sum_{i=1}^{n} \frac{1}{\sqrt{n}}|i\rangle|0\rangle \xrightarrow{I\otimes O_x} \sum_{i=1}^{n} \frac{1}{\sqrt{n}}|i\rangle|X(i)\rangle, \tag{59}$$

which means we can implement the oracle to X with one query to O_x. Hence, the approximate counting problem can be reduced to the mean estimation problem.

A k-parallel query call to x is

$$O_x^{\otimes k}\,|i_1,\ldots,i_k,b_1,\ldots,b_k\rangle = |i_1,\ldots,i_k,b_1\oplus x_{i_1},\ldots,b_k\oplus x_{i_k}\rangle \tag{60}$$

[4] proved a k-parallel query lower bound of the approximate counting problem.

▶ **Theorem 22** ([4]). *For any quantum query algorithm and boolean string $x \in \{0,1\}^n$,*

$$\Omega\left(\frac{\binom{n-|x|}{\epsilon n}\binom{|x|+\epsilon n}{|x|}}{k\binom{n-|x|-1}{\epsilon n-1}\binom{|x|+\epsilon n-1}{|x|}}\right) = \Omega\Big(\frac{p(1-p)}{\epsilon^2 k}\Big) \tag{61}$$

k-parallel queries to O_x is necessary to estimate $p = \frac{|x|}{n}$ to within additive error ϵ.

By Theorem 22, if we want to use constant k-parallel queries to estimate p up to additive error ϵ, k needs to satisfy

$$\frac{p(1-p)}{\epsilon^2 k} = O(1), \tag{62}$$

which means

$$k = \Omega\Big(\frac{p(1-p)}{\epsilon^2}\Big). \tag{63}$$

Now we give a sample complexity lower bound of algorithms solving Task 8 with $m = 1$ using Theorem 21. The difficulty of directly applying Theorem 21 is that it requires $\dim \operatorname{Im}(\Pi_c)$ to be small. To resolve it, we prove that any quantum mean estimator can be modified to recover the state in query register Q to $|\mathbf{0}\rangle$ with a small overhead so that correct answers lie in a much smaller subspace.

▶ **Theorem 23.** *Suppose all random variables in Task 8 have variance bounded by σ^2, and $|\mu| \le R$. Let $\mathcal{A}$ be a quantum query algorithm acting on registers Q, W solving the quantum non-identical mean estimation problem defined in Task 8 with repetition parameter $m = 1$ and accuracy $\epsilon/2$. Suppose that $\frac{1}{2}n_Q > n_W + 2\log\big(\frac{2R}{\epsilon}\big) + 1$, then it requires $T = \Omega(\frac{\sigma^2}{\epsilon^2})$ for the existence of such an algorithm $\mathcal{A}$, and $\mathcal{A}$ needs $T = \Omega\big(\frac{\sigma^2}{\epsilon^2}\big)$ quantum experiments.*

Proof. Use the uncomputation trick to combine Theorem 21 and Theorem 22. See [14] appendix B.3. ◀

4.1.3 Implication for Quantum Linear Systems

As mentioned in the introduction, we can possibly estimate A by the following procedure.

For fixed integers $t_0, \gamma = \Theta(\log(\sqrt{n}/\delta))$ and any $0 \leq t < n$, suppose we have a register storing $|\psi_{t_0+2\gamma t}\rangle$. We measure $|\psi_{t_0+2\gamma t}\rangle$ to obtain a classical state $x_{t_0+2\gamma t}$, and get $|\psi_{t_0+2\gamma t+1}\rangle$ as the second register of $U_f|\psi_{t_0+2\gamma t}\rangle|0\rangle$ (note that $|\psi_{t_0+2\gamma t}\rangle$ has collapsed after the measurement), which encodes the randomness of $x_{t_0+2\gamma t+1}$ given $x_{t_0+2\gamma t}$. Similarly, we can also obtain $|\psi^{-1}_{t_0+2\gamma t+1}\rangle$ by querying U_f^{-1}. After that, we compute $U_f|\psi_{t_0+2\gamma t+1}\rangle|0\rangle$ and collect the second register as $|\psi_{t_0+2\gamma t+2}\rangle$, and do this computation for all $t_0 + 2\gamma t + 1$ to $t_0 + 2(\gamma+1)t - 1$. Then we let $t = t + 1$ and repeat this process.

After such process, we have n classical samples at even steps $X_{t_0} := [x_{t_0}, x_{t_0+2\gamma}, \ldots, x_{t_0+2n\gamma-2}] \in \mathbb{R}^{n\times n}$, and n quantum samples at odd steps. It holds that X_{t_0} is full rank with probability 1 given that

$$AX_{t_0} = [x_{t_0+1}, \ldots, x_{t_0+2n\gamma-1}] + W_{t_0} + Z_{t_0} \tag{64}$$

where W_{t_0} is a zero-mean noise matrix and $\|Z_{t_0}\|_F \leq O(\delta)$. The matrix Z_{t_0} denotes the difference between $\mathbb{E}[x_{t_0+2\gamma t+1} \mid x_{t_0+2\gamma t}]$ and $\mathbb{E}[x_{t_0+2\gamma t+1} \mid x_{t_0+2\gamma t}, x_{t_0+2\gamma(t+1)}]$, which are close since $\|A^n\|_2 = O(-\exp(n))$. We define the quantum unitary U_{t_0} as

$$U_{t_0}|0\rangle := \int_W \sqrt{f_{t_0}(W)}|\psi_{t_0+1}, \ldots, \psi_{t_0+2n\gamma-1}\rangle X_{t_0}^{-1} \mathrm{d}W \tag{65}$$

where $f_{t_0}(W)$ is the pdf of random matrix $W_{t_0}X_{t_0}^{-1}$. Then we can use the quantum samples collected at steps $t_0+1, \ldots, t_0+2n\gamma-1$ as the return of query to U_{t_0} (or $U_{t_0}^{-1}$). Note that the mean of the random variable encoded by U_{t_0} is $O(\delta)$-close to A in Frobenius norm according to (64). However, the distribution encoded in U_{t_0} are different for different t_0 since X_{t_0} are different. The lower bound presented in the previous section shows that this methods cannot achieve a desired quantum speed-up since the oracle U_{t_0} can only be queried once for each t_0.

4.2 Lower Bounds for $m \geq 1$

Given a boolean string $|x| \in \{0,1\}^n$ and $k \in [n]$, the task of distinguishing $|x| = k$ and $|x| = k+1$ or $|x| = k-1$ can be reduced to estimating $\frac{|x|}{n}$ to within $\frac{1}{n}$ additive error, which can be regarded as a mean estimation problem. Therefore, the query complexity lower bound for the first problem is also a lower bound for the second problem. As a result, we first prove the query complexity lower bound of the first problem given non-identical oracles.

We use the same quantum query algorithm model in Section 4.1, where the algorithm pre-determines $U_0, \ldots, U_T$ and needs to distinguish the cases between $|x| = k$ and $|x| = k+1$ or $k-1$ for any $1 \leq k < n$.

▶ **Lemma 24.** *Given a sequence of oracles $O_{x_1}, \ldots, O_{x_T}$ encoding boolean strings $x_1, \ldots, x_T$ in $\{0,1\}^n$, suppose all strings have the same Hamming weight w and the algorithm can query each oracle at most m times in turn. For any $1 \leq k < n$ and $m = O(\sqrt{n})$, any quantum algorithm needs $\Omega(\frac{n}{m})$ queries in total to distinguish between $w = k$ and $w = k-1$ or $k+1$ with high probability.*

Proof. See [14] appendix B.4. ◀

Now we give a sample complexity lower bound of the quantum non-identical mean estimation problem with repetition parameter m.

▶ **Theorem 25.** *Suppose all random variables in Task 8 are Bernoulli random variables with mean $\mu \in (0,1)$ such that $\epsilon \leq \mu(1-\mu)$ and $\epsilon = O(\frac{1}{m^2})$. It requires $T = \Omega(\frac{1}{\epsilon m^2})$ if there exists a quantum algorithm which queries each random variable at most m times in turn solves this problem. Any such quantum query algorithm needs $mT = \Omega(\frac{1}{\epsilon m})$ quantum experiments in total.*

Proof. Let $n = \frac{1}{\epsilon}$ and $k = \mu n$. Since $\epsilon \leq \mu(1-\mu)$, we have $1 \leq k \leq n-1$. Given a boolean string $|x| \in \{0,1\}^n$, the task of distinguishing $|x| = k$ and $|x| = k+1$ or $|x| = k-1$ can be reduced to estimating $\frac{|x|}{n}$ to within $\frac{1}{n}$ additive error. The latter problem can be regarded as estimating the mean of a Bernoulli random variable X to within additive error $\epsilon = \frac{1}{n}$. Since one query to O_X can be implemented by one query to O_x, the query complexity lower bound for the first problem is also a lower bound for the second problem. From $\epsilon = O(\frac{1}{m^2})$, we have $m = O(\frac{1}{\sqrt{\epsilon}}) = O(\sqrt{n})$. Therefore, by Lemma 24, any quantum algorithm solving the quantum non-identical mean estimation problem with repetition parameter m needs $\Omega(\frac{1}{\epsilon m})$ quantum experiments in total. ◀

References

1 Charles H. Bennett. Time/space trade-offs for reversible computation. *SIAM Journal on Computing*, 18(4):766–776, 1989. doi:10.1137/0218053.

2 Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik: Progress of Physics*, 46(4-5):493–505, 1998.

3 Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002.

4 Paul Burchard. Lower bounds for parallel quantum counting. *CoRR*, abs/1910.04555, 2019. doi:10.48550/arXiv.1910.04555.

5 Shouvanik Chakrabarti, Rajiv Krishnakumar, Guglielmo Mazzola, Nikitas Stamatopoulos, Stefan Woerner, and William J. Zeng. A threshold for quantum advantage in derivative pricing. *Quantum*, 5:463, 2021. doi:10.22331/q-2021-06-01-463.

6 Arjan Cornelissen and Yassine Hamoudi. A sublinear-time quantum algorithm for approximating partition functions. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1245–1264. SIAM, 2023. doi:10.1137/1.9781611977554.ch46.

7 Paul Dagum, Richard Karp, Michael Luby, and Sheldon Ross. An optimal algorithm for Monte Carlo estimation. *SIAM Journal on Computing*, 29(5):1484–1496, 2000. doi:10.1137/S0097539797315306.

8 Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, 20(4):633–679, 2020. doi:10.1007/s10208-019-09426-y.

9 Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996. doi:10.1145/237814.237866.

10 Lov K. Grover and Jaikumar Radhakrishnan. Quantum search for multiple items using parallel queries. *arXiv preprint*, 2004. arXiv:quant-ph/0407217.

11 Yassine Hamoudi. Quantum sub-gaussian mean estimator. *CoRR*, abs/2108.12172, 2021. doi:10.48550/arXiv.2108.12172.

12 Yassine Hamoudi and Frédéric Magniez. Quantum chebyshev's inequality and applications. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, 2019.

13 Peter Hoyer, Troy Lee, and Robert Spalek. Negative weights make adversaries stronger. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 526–535, 2007.

14 Jiachen Hu, Tongyang Li, Xinzhao Wang, Yecheng Xue, Chenyi Zhang, and Han Zhong. Quantum non-identical mean estimation: Efficient algorithms and fundamental limits, 2024. arXiv:2405.12838.

15 Stacey Jeffery, Frederic Magniez, and Ronald De Wolf. Optimal parallel quantum query algorithms. *Algorithmica*, 79:509–529, 2017. doi:10.1007/s00453-016-0206-z.

16 Robin Kothari and Ryan O'Donnell. Mean estimation when you have the source code; or, quantum Monte Carlo methods. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1186–1215. SIAM, 2023. doi:10.1137/1.9781611977554.ch44.

17 Ashley Montanaro. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2181):20150301, 2015.

18 Michael A. Nielsen and Isaac L. Chuang. Quantum computation and quantum information. *Phys. Today*, 54(2):60, 2001.

19 Ben W Reichardt. Span programs and quantum query complexity: The general adversary bound is nearly tight for every boolean function. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 544–551. IEEE, 2009. doi:10.1109/FOCS.2009.55.

20 Max Simchowitz and Dylan Foster. Naive exploration is optimal for online LQR. In *International Conference on Machine Learning*, pages 8937–8948. PMLR, 2020. URL: http://proceedings.mlr.press/v119/simchowitz20a.html.

21 Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

22 Theodore J. Yoder, Guang Hao Low, and Isaac L. Chuang. Fixed-point quantum search with an optimal number of queries. *Physical review letters*, 113(21):210501, 2014.

23 Christof Zalka. Grover's quantum searching algorithm is optimal. *Physical Review A*, 60(4):2746, 1999.

24 Zhicheng Zhang, Qisheng Wang, and Mingsheng Ying. Parallel quantum algorithm for Hamiltonian simulation. *Quantum*, 8:1228, 2024. doi:10.22331/q-2024-01-15-1228.

Guidable Local Hamiltonian Problems with Implications to Heuristic Ansatz State Preparation and the Quantum PCP Conjecture

Jordi Weggemans
CWI & QuSoft, Amsterdam, The Netherlands
Fermioniq, Amsterdam, The Netherlands

Marten Folkertsma
CWI & QuSoft, Amsterdam, The Netherlands

Chris Cade
Fermioniq, Amsterdam, The Netherlands
QuSoft & University of Amsterdam (UvA), The Netherlands

Abstract

We study "Merlinized" versions of the recently defined Guided Local Hamiltonian problem, which we call "*Guidable* Local Hamiltonian" problems. Unlike their guided counterparts, these problems do not have a guiding state provided as a part of the input, but merely come with the promise that one *exists*. We consider in particular two classes of guiding states: those that can be prepared efficiently by a quantum circuit; and those belonging to a class of quantum states we call *classically evaluatable*, for which it is possible to efficiently compute expectation values of local observables classically. We show that guidable local Hamiltonian problems for both classes of guiding states are QCMA-complete in the inverse-polynomial precision setting, but lie within NP (or NqP) in the constant precision regime when the guiding state is classically evaluatable.

Our completeness results show that, from a complexity-theoretic perspective, classical Ansätze selected by classical heuristics are just as powerful as quantum Ansätze prepared by quantum heuristics, as long as one has access to quantum phase estimation. In relation to the quantum PCP conjecture, we (i) define a complexity class capturing quantum-classical probabilistically checkable proof systems and show that it is contained in $\mathsf{BQP}^{\mathsf{NP}[1]}$ for constant proof queries; (ii) give a no-go result on "dequantizing" the known quantum reduction which maps a QPCP-verification circuit to a local Hamiltonian with constant promise gap; (iii) give several no-go results for the existence of quantum gap amplification procedures that preserve certain ground state properties; and (iv) propose two conjectures that can be viewed as stronger versions of the NLTS theorem. Finally, we show that many of our results can be directly modified to obtain similar results for the class MA.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory

Keywords and phrases Quantum complexity theory, local Hamiltonian problem, quantum state ansatzes, QCMA, quantum PCP conjecture

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.10

Related Version *Full Version*: https://arxiv.org/abs/2302.11578 [48]

Funding CC was supported by Fermioniq B.V., and thanks QuSoft and CWI for their accommodation whilst this work was completed. MF and JW were supported by the Dutch Ministry of Economic Affairs and Climate Policy (EZK), as part of the Quantum Delta NL programme.

Acknowledgements The authors thank Jonas Helsen and Harry Buhrman for useful discussions and Sevag Gharibian and François Le Gall for their comments on an earlier version of the manuscript. We also thank Ronald de Wolf for providing feedback on the introduction. We thank the anonymous reviewers for their helpful comments, and in particular an anonymous STOC reviewer who provided an annotated version of an earlier version of this manuscript with very detailed and helpful comments.

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 10; pp. 10:1–10:24
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Quantum chemistry and quantum many-body physics are generally regarded as two of the most promising application areas of quantum computing [1,12]. Whilst perhaps the original vision of the early pioneers of quantum computing was to simulate the *time-dynamics* of quantum systems [13,26], for many applications one is interested in *stationary* properties. One particularly noteworthy quantity is the *ground state energy* (which corresponds to the smallest eigenvalue) of a local Hamiltonian describing a quantum mechanical system of interest, say a small molecule or segment of material. The precision to which one can estimate the ground state energy plays a crucial role in practice: for instance, in chemistry the relative energies of molecular configurations enter into the exponent of the term computing reaction rates, making the latter exceptionally sensitive to small (non-systematic) errors in energy calculations. The problem of estimating the smallest eigenvalue of a local Hamiltonian up to some additive error relative to the operator norm (the decision variant of which is known as the *local Hamiltonian problem*) is well-known to be QMA-hard when the required accuracy scales inversely with a polynomial. Therefore, it is generally believed that, without any additional help or structure, quantum computers are not able to accurately estimate the smallest eigenvalues of general local Hamiltonians, and there is some evidence that this hardness carries over to those Hamiltonians relevant to chemistry and materials science [40]. A natural question to ask is then the following: how much "extra help" needs to be provided in order to accurately estimate ground state energies using a quantum computer?

In the quantum chemistry community, it is often suggested that this extra help could come from a classical heuristic that first finds some form of *guiding state*: a classical description of a quantum state that can be used as an input to a quantum algorithm to compute the ground state energy accurately [38]. Concretely, this comes down to the following two-step procedure [17]:

- Step 1 (Guiding state preparation): A classical heuristic algorithm is applied to obtain a *guiding state* $|\psi\rangle$, which is hoped to have "good"[1] fidelity with the ground space.
- Step 2: (Ground state energy approximation): The guiding state $|\psi\rangle$ is used as input to Quantum Phase Estimation (QPE) to efficiently and accurately compute the corresponding ground state energy.

Step 2 of the above procedure can be formalised by the *Guided k-local Hamiltonian problem (k-GLH)*, which was introduced in [28] and shown to be BQP-complete under certain parameter regimes that were subsequently improved and tightened in [17]. The problem k-GLH is stated informally as follows: given a k-local Hamiltonian H, an appropriate classical "representation" of a guiding state $|u\rangle$ promised to have ζ-fidelity with the ground space of H, and real thresholds $b > a$, decide if the ground state energy of H lies above or below the interval $[a, b]$.

In a series of works [17,18,28], it was shown that 2-GLH is BQP-complete for *inverse polynomial* precision and fidelity, i.e. $b - a \geq 1/\text{poly}(n)$ and $\zeta = 1 - 1/\text{poly}(n)$ respectively. In contrast, when $b - a \in \Theta(1)$ and $\zeta = \Omega(1)$, k-GLH can be efficiently solved *classically* by using a dequantized version of the quantum singular value transformation [28].

The GLH problem forms the starting point of this work. We study "*Merlinized*" versions of GLH – in which guiding states are no longer given as part of the input but instead are only promised to exist – and use these as a way to gain some insight into important theoretical questions in quantum chemistry and complexity theory. In the subsequent paragraphs, we introduce some of the motivating questions guiding the study of the complexity of these so-called "guidable" local Hamiltonian problems.

[1] "Good" here means at least inverse polynomial in the number of qubits the Hamiltonian acts on.

Ansätze for state preparation. Step 1 of the aforementioned two-step procedure generally requires one to have access to classical heuristics capable of finding guiding states whose energies can be estimated classically (as a metric to test whether candidate states are expected to be close to the actual ground state or not). Furthermore, these "trial states" should also be preparable as quantum states on a quantum computer, so that they can be used as input to phase estimation in Step 2. In [28], inspired by a line of works that focused on the dequantization of quantum machine learning algorithms [20, 33, 44], a particular notion of "sampling-access" to the guiding state $|u\rangle$ is assumed. Specifically, it is assumed that one can both query the amplitude of arbitrary basis states, and additionally that one can sample basis states according to their l_2 norm with respect to the overall state $|u\rangle$. This can be a somewhat powerful model [22], and it is closely related to the assumption of QRAM access to classical data, and thus in the context of quantum machine learning (where such access is commonly assumed), it makes sense to compare quantum machine learning algorithms to classical algorithms with sampling access to rule out quantum speed-ups that come merely from having access to quantum states that are constructed from exponential-size classical data. However, for quantum chemistry and quantum many-body applications, this type of access to quantum states seems to be somewhat artificial. From a theoretical perspective, one might wonder to what extent this sampling access model "hides" some complexity, allowing classical algorithms to perform well on the problem when they otherwise would not.

Moreover, one may ask whether the fact that the ground state preparation in Step 1 considers only *classical* heuristics might be too restrictive. *Quantum* heuristics for state preparation, such as variational quantum eigensolvers [46] and adiabatic state preparation techniques [8], have received considerable attention as possible quantum approaches within the NISQ era, and one can argue that even in the fault-tolerant setting, such heuristics will likely still be viable approaches to state preparation, in particular when used in conjunction with Quantum Phase Estimation.

The quantum PCP conjecture. Arguably the most fundamental result in classical complexity theory is the Cook-Levin Theorem [21, 36], which states that constraint satisfaction problems (CSPs) are NP-complete. The PCP theorem [10, 11], which originated from a long line of research on the complexity of interactive proof systems, can be viewed as a "strengthening" of the Cook-Levin theorem. In its proof-checking form, it states that all decision problems in NP can be decided, with a constant probability of error, by only checking a constant number of bits of a polynomially long proof string y (selected randomly from the entries of y). There are also alternative equivalent formulations of the PCP theorem. One is in terms of *hardness of approximation*: it states that it remains NP-hard to decide whether an instance of CSP is either completely satisfiable, or whether no more than a constant fraction of its constraints can be satisfied.[2] Naturally, quantum complexity theorists have proposed proof-checking and hardness of approximation versions of PCP in the quantum setting. Given the close relationship between QMA and the local Hamiltonian problem, the most natural formulation is in terms of hardness of approximation: in this context, the *quantum* PCP conjecture roughly states that energy estimation of a (normalized) local Hamiltonian up to *constant* precision, relative to the operator norm of the Hamiltonian, remains QMA-hard. This conjecture is arguably one of the most important open problems in quantum complexity theory and has remained unsolved for nearly two decades.

[2] The transformation of a CSP to another one which is hard to approximate is generally referred to as *gap amplification*, and is realised in Dinur's proof of the PCP theorem [24].

One way to shed light on the validity of the quantum PCP conjecture can be to study PCP-type conjectures for other “Merlinized” complexity classes. Up until this point, PCP-type conjectures have not been considered for other classes besides NP and QMA.[3] However, there is the beautiful result of [7], which studies the possibility of a gap amplification procedure for the class MA by considering a particular type of Hamiltonian: uniform stoquastic local Hamiltonians. The authors show that deciding whether the energy of such a Hamiltonian is exactly zero or inverse polynomially bounded away from zero is MA-hard, but that the problem is in NP when this interval is increased to be some constant. Consequently, this implies that there can exist a gap-amplification procedure for uniform stoquastic Local Hamiltonians (in analogy to the gap amplification procedure for constraint satisfaction problems in the original PCP theorem) if and only if MA = NP – i.e. if MA can be derandomized. Since MA $\subseteq$ QMA, this result also shows that if a gap amplification procedure for the general local Hamiltonian problem would exist that “preserves stoquasticity”, then it could also be used to derandomize MA.

1.1 Summary of main results

1.1.1 Completeness results for guidable local Hamiltonian problems

Inspired by classical heuristics that work with Ansätze to approximate the ground states of local Hamiltonians, we define a general class of states that we call *classically evaluatable and quantumly preparable.*

▶ **Definition 1** (Classically evaluatable and quantumly preparable states). *We say that an n-qubit state $|u\rangle$ is ϵ-classically evaluatable if*

(i) *it has an efficient classical description which requires at most a polynomial number of bits to write down and*

(ii) *one can, given such a description classically efficiently compute expectation values of $\mathcal{O}(\log n)$-local observables of $|u\rangle$ up to precision ϵ and with probability $\geq 1 - 1/\text{poly}(n)$.*

In addition, we say that the state is also quantumly preparable if (iii) there exists a quantum circuit that prepares $|u\rangle$ using only a polynomial number of two-qubit gates. Furthermore, if $\epsilon = 0$ the algorithm in (ii) is deterministic instead of probabilistic and we simply say that $|u\rangle$ is classically evaluatable.

This definition of states is very closely related to the definition of query and sampling access to quantum states given by Gharibian and Le Gall [28], which slightly generalizes the original definition as first proposed by Tang used to dequantize quantum algorithms for recommendation systems [44]. There are three main motivations for introducing this new class of states:

1. It seems rather difficult to find Ansätze that are used in practice for ground state energy estimation that satisfy all conditions of query and sampling access. As one of the main motivations of this work is to investigate the power of quantum versus classical state preparation when one has access to Quantum Phase Estimation, we want to define a class of states that can both be prepared efficiently on a quantum computer and which contains a large class of Ansätze commonly used in practice.

[3] Barring a result by Drucker which proves a PCP theorem for the class AM [25]; though there is no direct relationship between QMA and AM and hence it is not clear whether this gives any intuition about the likely validity of the quantum PCP conjecture.

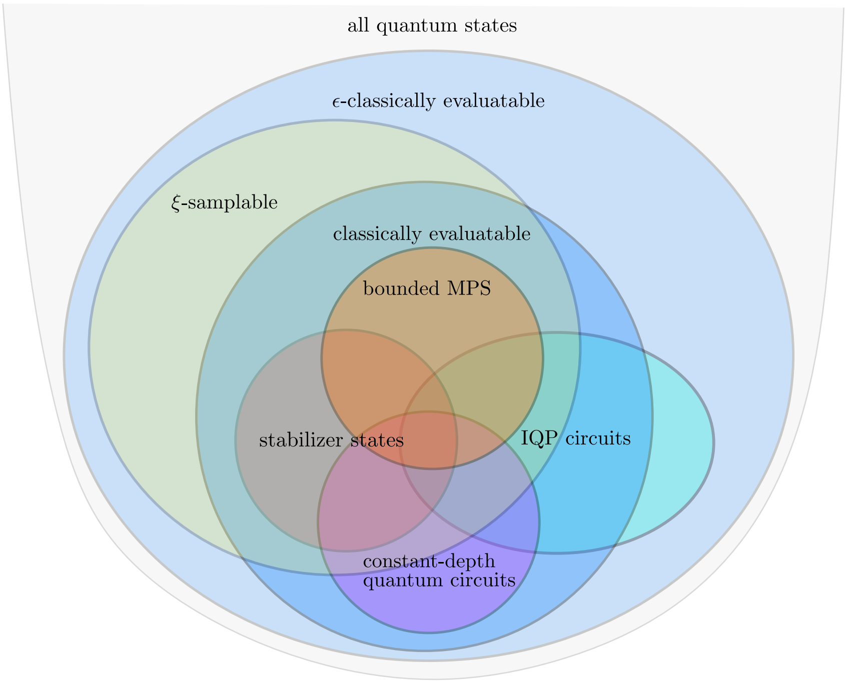


Figure 1 Visualization of the (conjectured) relations between classes of quantum states considered in this work, given a Hilbert space of a fixed dimension. For MPS, we only consider states with polynomially-bounded bond and local dimension. We take $\xi \leq \epsilon/8 \leq 1/3$, such that by Theorem 2 we have that (i) all ξ-samplable states are also ϵ-classically evaluatable and (ii) constant-depth and IQP circuits are not ξ-samplable.

2. Analogous to Dinur's construction, one would expect that determining if a local Hamiltonian has ground state energy (exponentially close to) zero or some constant away from zero is QMA-hard if the quantum PCP conjecture is true. However, there are arguments from physics[4] as to why one might expect this problem to be in NP [41]. To study the question of containment in NP it is necessary to be able to work with states within a deterministic setting, and therefore it does not make sense to rely on a form of sampling access which inherently relies on a probabilistic model of computation.
3. To add to the previous point, being able to study containment in NP comes with the additional advantage of being able to make statements about whether the problem admits a PCP by the classical PCP theorem. No such theorem is currently known for MA.

To strengthen the first point we find four concrete examples of Ansätze that satisfy all three conditions: matrix product states (MPS), stabilizer states, constant-depth quantum circuits and IQP circuits [15]. Explicit definitions of these classes of states as well as proofs of containment can be found in [48]. The first two examples are in fact also perfectly samplable. However, constant-depth quantum circuits are not even approximately samplable (under the conjecture that BQP $\not\subset$ AM [45]). We can formalize this in the following theorem which relates ξ-samplable states to ξ-classically evaluatable states

[4] In this setting the LH problem becomes equivalent to determining whether the free energy of the system becomes negative at a finite temperature. One expects then that at such temperatures, the system loses its quantum characteristics on the large scale, making the effects of long-range entanglement become negligible. Hence, this means that the ground state of such a system should have some classical description, which places the problem in NP [9].

▶ **Theorem 2.** *For any $\xi > 0$, any ξ-samplable state is also $\mathcal{O}(\xi)$-classically evaluatable. On the other hand, there exist states that are perfectly classically evaluatable but not ξ'-samplable for all $0 < \xi' < 1/3$, unless $\mathsf{BQP} \subseteq \mathsf{AM}$.*

The proof of this theorem can be found in the full version of this paper [48]. This theorem gives rise to a (conjectured) hierarchical structure of states as depicted in Figure 1. For the remainder of our work, we will focus on (0-)classically evaluatable states, which by Definition 1 means that there exists a deterministic classical algorithm for computing expectation values. A notable advantage of this approach is, as opposed to 0-samplable states, that this allows us to give NP containment results.

Our main focus is on a new family of Hamiltonian problems, in which we are promised that the ground state is close (with respect to fidelity) to some state from a particular class of states, called *guiding states*. We make a distinction between different types of promises one can make with respect to the existence of guiding states: we either assume that the guiding states are of the form of Definition 1 (with or without the promise that the states are also quantumly preparable), or that there exists an efficient quantum circuit that prepares the guiding state.

▶ **Definition 3** (Guidable Local Hamiltonian problems)**.** *Guidable Local Hamiltonian Problems are problems defined by having the following input, promise, output and some extra promise to be precisely defined below for each of the problems separately:*

Input: *A k-local Hamiltonian H with $\|H\| \leq 1$ acting on n qubits, threshold parameters $a, b \in \mathbb{R}$ such that $b - a \geq \delta > 0$ and a fidelity parameter $\zeta \in (0, 1]$.*

Promise: *We have that either $\lambda_0(H) \leq a$ or $\lambda_0(H) \geq b$ holds, where $\lambda_0(H)$ denotes the ground state energy of H.*

Extra promises: *Let Π_{gs} be the projection on the subspace spanned by the ground states of H. Then for each problem, we have that either one of the following promises holds:*

1. *There exists a classically evaluatable state $u \in \mathbb{C}^{2^n}$ for which $\|\Pi_{gs} u\|^2 \geq \zeta$. Then the problem is called the* ***Classically Guidable Local Hamiltonian Problem****, shortened as* $\mathsf{CGaLH}(k, \delta, \zeta)$. *If $|u\rangle$ is also quantumly preparable, we call the problem the* ***Classically Guidable and Quantumly Preparable Local Hamiltonian Problem****, shortened as* $\mathsf{CGaLH}^*(k, \delta, \zeta)$.
2. ***Quantumly Guidable k-LH*** ($\mathsf{QGaLH}(k, \delta, \zeta)$)*: There exists a quantum circuit of polynomially many two-qubit gates that produces the state $|\phi\rangle$ for which $\|\Pi_{gs} |\phi\rangle\|^2 \geq \zeta$.*

Output:
- *If $\lambda_0(H) \leq a$, output* YES.
- *If $\lambda_0(H) \geq b$, output* NO.

We note that a guidable local Hamiltonian problem variant for a different class of guiding states was already introduced in Section 5 of [28] without giving any hardness results. Using techniques from Hamiltonian complexity we obtain the following completeness results.[5]

▶ **Theorem 4** (Complexity of guidable local Hamiltonian problems)**.** *For $k = 2$ and $\delta = 1/\text{poly}(n)$, we have that both $\mathsf{CGaLH}^*(k, \delta, \zeta)$ and $\mathsf{QGaLH}(k, \delta, \zeta)$ are QCMA-complete when $\zeta \in (1/\text{poly}(n), 1 - 1/\text{poly}(n))$.*

A basic version of the hardness proof can be found in Section 3.1, with the remainder written down in the full version [48]. A direct corollary of the above theorem is the following.

[5] In fact $\mathsf{QGaLH}(k, \delta, \zeta)$ remains QCMA-hard all the way up to $\zeta = 1$.

▶ **Corollary 5** (Classical versus quantum state preparation). *When one has access to a quantum computer (and in particular quantum phase estimation), then having the ability to prepare any quantum state preparable by a polynomially-sized quantum circuit is no more powerful than the ability to prepare states from the family of classically evaluatable and quantumly preparable states, when the task is to decide the local Hamiltonian problem with precision* $1/\text{poly}(n)$.

It should be noted that our result does *not* imply that all Hamiltonians which have efficiently quantumly preparable guiding states also necessarily have guiding states that are classically evaluatable. All this result says is that for any instance of the guidable local Hamiltonian problem with the promise that there exist guiding states that can be efficiently prepared by a quantum computer, there exists an (efficient) *mapping* to another instance of the guidable local Hamiltonian problem with the promise that there exist guiding states that are classically evaluatable and quantumly preparable. Whilst this reduction is efficient in the complexity-theoretic sense, it might not be for practical purposes, as it would likely remove all the physical structure present in the original Hamiltonian. Hence, the main implication of our result is not that these kinds of reductions are of practical merit, but that at least from a complexity-theoretic point of view the aforementioned classical-quantum hybrid approach of guiding state selection through *classical* heuristics combined with *quantum* energy estimation is at least as powerful as using quantum heuristics for state preparation instead.

We complement our quantum hardness results with classical containment results (of the classically guidable local Hamiltonian problem), obtained through a deterministic dequantized version of Lin and Tong's ground state energy estimation algorithm [37]. Here CGaLH is just as CGaLH* but without the promise of the guiding state being quantumly preparable.

▶ **Theorem 6** (Classical containment of the classically guidable local Hamiltonian problem). *Let* $k = \mathcal{O}(\log n)$. *When* δ *is constant, we have that* CGaLH(k, δ, ζ) *is in* NP *when* ζ *is constant and is in* NqP *when* $\zeta = 1/\text{poly}(n)$. *Here* NqP *is just as* NP *but with the Turing machine being allowed to run in quasi-polynomial time.*

Theorem 6 follows directly by applying the spectral amplification technique, as described in Section 3.2.

1.1.2 Quantum-classical probabilistically checkable proofs

We introduce the notion of a *quantum-classical probabilistically checkable proof system* in the following way.

▶ **Definition 7** (Quantum-Classical Probabilistically Checkable Proofs (QCPCP)). *Let* $n \in \mathbb{N}$ *be the input size and* $p, q : \mathbb{N} \to \mathbb{N}$, $c, s : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ *with* $c - s > 0$. *A promise problem* $A = (A_{yes}, A_{no})$ *has a* $(p(n), q(n), c, s)$-QCPCP*-verifier if there exists a quantum algorithm* V *which acts on an input* $|x\rangle$ *and a polynomial number of ancilla qubits, plus an additional bit string* $y \in \{0,1\}^{p(n)}$ *from which it is allowed to read at most* $q(n)$ *bits (non-adaptively), followed by a measurement of the first qubit, after which it accepts only if the outcome is* $|1\rangle$, *and satisfies:*

Completeness. *If* $x \in A_{yes}$, *then there is a* $y \in \{0,1\}^{p(n)}$ *such that the verifier accepts with probability at least* c,

Soundness. *If* $x \in A_{no}$, *then for all* $y \in \{0,1\}^{p(n)}$ *the verifier accepts with probability at most* s.

A promise problem $A = (A_{yes}, A_{no})$ *belongs to* QCPCP$[p, q, c, s]$ *if it has a* $(p(n), q(n), c, s)$-QCPCP *verifier. If* $p(n) = \mathcal{O}(\text{poly}(n))$, $c = 2/3$, *and* $s = 1/3$, *we simply write* QCPCP$[q]$.

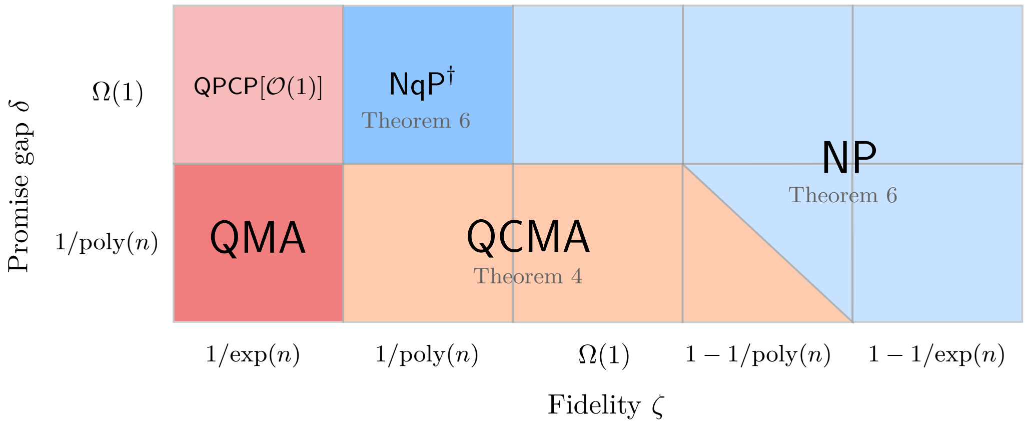


Figure 2 Complexity characterization of CGaLH$^*(k, \delta, \zeta)$ over parameter regime δ and ζ, for $k = \mathcal{O}(1)$. Any classification indicates completeness for the respective complexity class, except for NqP, for which we only know containment (indicated by the " † "). Here completeness for certain parameter combinations means that for all functions of the indicated form, the problem is contained in the complexity class, and for a subset of these functions the problem is also hard. The results for QPCP[$\mathcal{O}(1)$] and QMA follow directly from [4] and [35].

We remark that there are likely several ways to characterise a "quantum-classical PCP", with some being more or less natural than others. With that said, we believe that the above characterisation is well-motivated for the following reasons:

1. It is a natural definition following the structure of a QPCP verifier, now with proofs given as in the standard definition of QCMA. Moreover, one can show that the non-adaptiveness is not restrictive when the number of queries is constant (this is proved in the full version [48]).
2. QCPCP[$\mathcal{O}(1)$] captures the power of BQP as well as NP (via the PCP theorem), which are both believed to be strictly different complexity classes. Since techniques used to prove the PCP theorem are difficult (or impossible) to translate to the quantum setting [5], studying QCPCP[$\mathcal{O}(1)$] might provide a fruitful direction with which to obtain the first non-trivial lower bound on the complexity of QPCP[$\mathcal{O}(1)$]. Indeed, the currently best-known lower bound on the complexity of QPCP[$\mathcal{O}(1)$] is only NP via the PCP theorem.

Given this definition for QCPCPs, our "quantum-classical" PCP conjecture is naturally formulated as follows.

▶ **Conjecture 8** (quantum-classical PCP conjecture). *There exists a constant $q \in \mathbb{N}$ such that* QCMA = QCPCP[q].

If true, this conjecture would give a "QCMA lower bound" on the power of quantum PCP systems, showing that a PCP theorem holds for (quantum) classes above NP, taking a step towards proving the quantum PCP conjecture. If it is false, but the quantum PCP conjecture is true, then this suggests that QPCP systems must take advantage of the "quantumness" of their proofs to obtain a probabilistically checkable proof system. In particular, since QCMA $\subseteq$ QMA, this would imply the existence of a quantum PCP system for every problem in QCMA, but *not* a quantum-classical one, even though the problem admits a classical proof that can be efficiently verified when we are allowed to look at all of its bits.

Our main result regarding QCPCP[$\mathcal{O}(1)$] is that we can provide a non-trivial upper bound on the complexity of the class.

▶ **Theorem 9** (Upper bound on QCPCP, from Theorem 25). $\mathsf{QCPCP}[\mathcal{O}(1)] \subseteq \mathsf{BQP}^{\mathsf{NP}[1]}$.

Here $\mathsf{BQP}^{\mathsf{NP}[1]}$ is the class of all problems that can be solved by a BQP-verifier that makes a single query to an NP-oracle. The key idea behind the proof is that a quantum reduction can be used to transform a QCPCP verification circuit to a local Hamiltonian that is *diagonal* in the computational basis, and thus can be solved with a single query to an NP oracle.

An implication of Theorem 9 is that it can be used to show that under the assumption $\mathsf{NP} \subseteq \mathsf{BQP}$ and the quantum-classical PCP conjecture being true, we have that $\mathsf{PH} \subseteq \mathsf{BQP}$. This follows from the fact that $\mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{QCMA}$ and

$$\mathsf{NP}^{\mathsf{NP}} \subseteq \mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}^{\mathsf{NP}} \subseteq \mathsf{BQP}^{\mathsf{BQP}} = \mathsf{BQP},$$

where the first and the third "$\subseteq$" are by assumption, the second is by the assumption of Conjecture 8 to be true and the last equality follows from the fact that BQP is self-low. We then have that $\mathsf{PH} \subseteq \mathsf{BQP}$ follows by induction, just as is the case for BPP [50].[6] Moreover, this would also imply that under these assumptions $\mathsf{QCMA} \subseteq \mathsf{BQP}$, since

$$\mathsf{QCMA} \subseteq \mathsf{QCPCP}[\mathcal{O}(1)] \subseteq \mathsf{BQP}^{\mathsf{NP}} \subseteq \mathsf{BQP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}.$$

Both of these implications would provide further evidence that it is unlikely that $\mathsf{NP} \subseteq \mathsf{BQP}$. However, it is known that there exists an oracle relative to which $\mathsf{NP}^{\mathsf{BQP}^A} \not\subset \mathsf{BQP}^{\mathsf{NP}^A}$ [3]. Nevertheless, this does not necessarily mean the premise (i.e. the quantum-classical PCP conjecture) is false: one can also easily construct an oracle separation between PCP and NP, and both classes are now known to be equal [27]. However, this suggests that, if Conjecture 8 is true, showing so requires non-relativizing techniques, just as was the case for the PCP theorem.

1.1.3 Three implications for the quantum PCP conjecture

We use our obtained results on QCPCP and CGaLH to obtain two new results and a new conjecture with respect to the quantum PCP conjecture. First, we give evidence that it is unlikely that there exists a *classical* reduction from a QPCP-system (see [4] or [16] for a formal definition) to a local Hamiltonian problem with a constant promise gap having the same properties as the known *quantum* reduction (see for example [16, 31]), unless $\mathsf{BQP} \subseteq \mathsf{QCPCP}[\mathcal{O}(1)] \subseteq \mathsf{NP}$, something that is not expected to hold [2, 42].

▶ **Theorem 10** (No-go for classical polynomial-time reductions). *For any $\epsilon < 1/6$ there cannot exist a classical polynomial-time reduction from a $\mathsf{QPCP}[\mathcal{O}(1)]$ verification circuit V to a local Hamiltonian H such that, given a proof $|\psi\rangle$,*

$$|\mathbb{P}[V \text{ accepts } |\psi\rangle] - (1 - \langle\psi| H |\psi\rangle)| \leq \epsilon,$$

unless $\mathsf{QCPCP}[\mathcal{O}(1)] \subseteq \mathsf{NP}$ (which would imply $\mathsf{BQP} \subseteq \mathsf{NP}$).

The proof is given in the full version [48]. This provides strong evidence that allowing for reductions to be quantum is indeed necessary to show equivalence between the gap amplification and proof verification formulations of the quantum PCP conjecture [5].

Second, our classical containment results of CGaLH with constant promise gap can be viewed as no-go theorems for a gap amplification procedure for QPCP having certain properties, as illustrated by the following result.

[6] See also `https://blog.computationalcomplexity.org/2005/12/pulling-out-quantumness.html`.

▶ **Theorem 11** (No-go results for Hamiltonian gap amplification). *There cannot exist a polynomial time classical gap amplification procedure for the local Hamiltonian problem that preserves the fidelity between the ground space of the Hamiltonian and any classically evaluatable state up to a*

- *multiplicative constant, unless* QCMA = NP*, or*
- *multiplicative inverse polynomial, unless* QCMA $\subseteq$ NqP*.*

The theorem follows directly from Theorem 6. This result is analogous to the result of [7], which rules out a gap amplification procedure that preserves stoquasticity under the assumption that MA $\neq$ NP (or taking a different view, proving the existence of such gap amplifications would allow one to simultaneously prove that MA can be derandomized). Moreover, we point out that many Hamiltonian gadget constructions *do* satisfy such fidelity-preserving conditions, and indeed are precisely those that were used in [17] to improve the hardness results for the guided local Hamiltonian problem. We obtain similar results for the class MA by considering a variant of CGaLH that restricts the Hamiltonian to be stoquastic (See Appendix C in the full version [48]).

Third, we can use our results to formulate a stronger version of the NLTS theorem (and an alternative to the NLSS conjecture [28]), which we will call the *No Low-energy Classically evaluatable States conjecture.* This conjecture can hopefully provide a new stepping stone towards proving the quantum PCP conjecture.

▶ **Conjecture 12** (NLCES conjecture). *There exists a family of local Hamiltonians $\{H_n\}_{n\in\mathbb{N}}$ on n qubits, and a constant $\beta > 0$, such that for sufficiently large n for every classically evaluatable state $u \in \mathbb{C}^{2^n}$ as per Definition 1, we have that $\langle u| H_n |u\rangle \geq \lambda_0(H_n) + \beta$.*

Just as is the case for the NLSS conjecture and the NLTS theorem, the NLCES conjecture would, if proven to be true, not necessarily imply the quantum PCP conjecture. For example, it might be that there exist states that can be efficiently described classically but for which computing expectation values is hard (just as, for example, tensor network contraction is #P-hard in the worst case [14, 43]). Furthermore, as we have shown in this work, states with high energy but also a large fidelity with the ground state suffice as witnesses to decision problems on Hamiltonian energies, and these would not be excluded by a proof of the NLCES conjecture above. To make this more concrete, in the full version [48] we also formulate an even stronger version of the NLCES conjecture, which states that there must be a family of Hamiltonians for which no classically evaluatable state has good fidelity with the low energy spectrum.

2 Preliminaries

2.1 Notation

We write $\lambda_i(A)$ to denote the ith eigenvalue of a Hermitian matrix A, ordered in non-decreasing order, with $\lambda_0(A)$ denoting the smallest eigenvalue (ground state energy). When we write $\|\cdot\|$ we refer to the operator norm when its input is a matrix and Euclidean norm for a vector.

2.2 Complexity theory

All complexity classes will be defined with respect to promise problems. To this end, we take a (promise) problem $A = (A_{\text{yes}}, A_{\text{no}})$ to consist of two non-intersecting sets $A_{\text{yes}}, A_{\text{no}} \subseteq \{0,1\}^*$ (the YES and NO instances, respectively). We have that $A_{\text{inv}} = \{0,1\}^* \setminus A_{\text{yes}} \cup A_{\text{no}}$ is the set of all invalid instances, and we do not care how a verifier behaves on problem instances $x \in A_{\text{inv}}$ (i.e. it can accept or reject arbitrarily). We assume that the reader is familiar with the complexity classes used in this work, and else suggest reading the formal definitions in [48] or the complexity theory zoo (`https://complexityzoo.net/Complexity_Zoo`). However, since it is crucial to our construction, we will explicitly state the class UQCMA, which is just as QCMA but with a unique accepting witness in the YES-case.

▶ **Definition 13** (UQCMA). *A promise problem $A = (A_{yes}, A_{no})$ is in* UQCMA*[c,s] if and only if there exists a polynomial-time uniform family of quantum circuits $\{V_n\}$ and a polynomial p, where V_n takes as input a string $x \in \{0,1\}^*$ with $|x| = n$, and a $p(n)$-qubit witness quantum state $|\psi\rangle$ and decides on acceptance or rejection of x such that*

- *if $x \in A_{yes}$ then there exists a unique $y^* \in \{0,1\}^{p(n)}$ such that V_n accepts $(x, |y^*\rangle)$ with probability $\geq c$, and for all $y \neq y^*$ we have that V_n accepts $(x, |y\rangle)$ with probability $\leq s$;*
- *if $x \in A_{no}$ then for every witness state $y \in \{0,1\}^{p(n)}$, V_n accepts $(x, |y\rangle)$ with probability $\leq s$,*

where $c - s = 1/\text{poly}(n)$. If $c = 2/3$ and $s = 1/3$, we abbreviate to UQCMA*.*

In [6] it was shown that there exists a randomized reduction from QCMA to UQCMA, analogous to the Valiant-Vazirani theorem for NP [47].

Oracle access. For a (promise) class $\mathcal{C}$ with complete (promise) problem A, the class $\mathsf{P}^{\mathcal{C}} = \mathsf{P}^{A}$ is the class of all (promise) problems that can be decided by a polynomial-time verifier circuit V with the ability to query an oracle for A. If V makes invalid queries (i.e. $x \in A_{\text{inv}}$), the oracle may respond arbitrarily. However, since V is deterministic, it is required to output the same final answer regardless of how such invalid queries are answered [29, 30]. Hence, the answer to any query outside of the promise set should not influence the final output bit. For a function f, we define $\mathsf{P}^{\mathcal{C}[f]}$ to be just as $\mathsf{P}^{\mathcal{C}}$ but with the additional restriction that V may ask at most $f(n)$ queries on an input of length n. One defines $\mathsf{NP}^{\mathcal{C}}$ or $\mathsf{NP}^{\mathcal{C}[f]}$ in the same way but replacing the polynomial-time deterministic verifier V by a nondeterministic polynomial-time verifier V', taking an additional input $y \in \{0,1\}^{p(n)}$ for some polynomial $p(n)$.

3 (Partial) proofs of a selection of results

In this section we will give some of the key lemmas and theorems which are behind the results presented in Section 1. The full proofs as well as in-depth discussions can be found in the full version [48].

3.1 QCMA-completeness of guidable local Hamiltonian problems

We prove a basic version of the reduction that shows that Guidable Local Hamiltonian problems are QCMA-hard in the inverse polynomial precision regime. Our construction is based on a combination of the ideas needed to show BQP-hardness for the Guided Local Hamiltonian problem [17, 18, 28] and the small penalty clock construction of [23].

The first obstruction one encounters in adopting the ideas from the BQP-hardness proofs of the Guided Local Hamiltonian problem to the guidable setting is the fact that QCMA verifiers, unlike BQP, have a proof register. In QCMA the promises of completeness and soundness are always with respect to computational basis state witnesses. Hence, these might no longer hold when *any* quantum state can be considered as witness: for example, in the NO-case there might be highly entangled states which are accepted with probability

$\geq 2/3$. When considering a circuit problem, the verifier can easily work around this by simply measuring the witness and then proceeding to verify with the resulting computational basis state. However, there is also another trick, which retains the unitarity of the verification circuit – and which we will denote as the "CNOT-trick" from now on – to force the witness to be classical, first used in proving QCMA-completeness of the *Low complexity low energy states* problem in [49].

▶ **Lemma 14** (The "CNOT-trick"). *Let $p(n) : \mathbb{N} \to \mathbb{R}_{>0}, q(n) : \mathbb{N} \to \mathbb{R}_{>0}$ be polynomials. Let U_n be a quantum polynomial-time verifier circuit that acts on an n-qubit input register A, a $p(n)$-qubit witness register B and a $q(n)$-qubit workspace register C, initialized to $|0\rangle^{\otimes q(n)}$. Denote Π_0 for the projection on the first qubit being zero. Let Q be the Marriott-Watrous operator of the circuit, defined as*

$$Q = \left(\langle x| \otimes I_w \otimes \langle 0|^{\otimes q(n)}\right) U_n^\dagger \Pi_0 U_n \left(|x\rangle \otimes I_w \otimes |0\rangle^{\otimes q(n)}\right). \tag{1}$$

Consider yet another additional $p(n)$-qubit workspace D initialized to $|0\rangle^{\otimes p(n)}$, on which U_n does not act. Then by prepending U_n with $p(n)$ CNOT-operations, each of which is controlled by a single qubit in register B and targeting the corresponding qubit in register D, the corresponding Marriott-Watrous operator becomes diagonal in the computational basis.

The corresponding lemma and proof can be found in the full version [48]. The next obstruction one faces is that in the QCMA setting there might be multiple proofs which all have exponentially close, or even identical, acceptance probabilities. The analysis of the BQP-hardness proof fails to translate directly to this setting, and another technique is needed. For this, we resort to (i) using the fact that QCMA is equal to UQCMA under randomized reductions and (ii) use a *small-penalty clock construction* of [23]. The key idea is to use a Feynman-Kiteav circuit-to-Hamiltonian mapping modified with a tunable parameter ϵ, which maps a quantum verification circuit U_n, consisting of T gates from a universal gate set of at most 2-local gates, taking input x and a quantum proof $|\psi\rangle \in \left(\mathbb{C}^2\right)^{\otimes \text{poly}(n)}$ to a k-local Hamiltonian of the form

$$H_{FK}^x = H_{\text{in}} + H_{\text{clock}} + H_{\text{prop}} + \epsilon H_{\text{out}}. \tag{2}$$

The value of k depends on the used construction. Intuitively, the first three terms check that the Hamiltonian is faithful to the computation and the last term shifts the energy level depending on the acceptance probability of the circuit. Just as in [23], we will use Kempe and Regev's 3-local construction. A precise description of the individual terms in (2) can be found in [34], and will not be relevant for our work, except for the fact that H_{FK}^x has a polynomially bounded operator norm. The ground state of the first three terms $H_0 = H_{\text{in}} + H_{\text{clock}} + H_{\text{prop}}$ is given by the so-called *history state*, which is given in [34] by

$$|\eta(\psi)\rangle = \frac{1}{\sqrt{T+1}} \sum_{t=0}^{T} U_t \dots U_1 |\psi\rangle |0\rangle \left|\hat{t}\right\rangle, \tag{3}$$

where $|\psi\rangle$ is the quantum proof and $\hat{t}$ the unary representation of the time step of the computation given by

$$\hat{t} = |\underbrace{1\dots1}_{t}\underbrace{0\dots0}_{T-t}\rangle.$$

From the construction in [34], it is easily verified that if U_n accepts $(x, |\psi\rangle)$ with probability p then we have that the corresponding history state has energy

$$\langle\eta(\psi)| H^x_{FK} |\eta(\psi)\rangle = \epsilon\frac{1-p}{T+1}. \tag{4}$$

Though the core idea behind the small-penalty clock construction is identical to the one used in the BQP-hardness proof – rescaling the weight of the H_{out} term as compared to the other terms in a Feynman-Kiteav circuit-to-Hamiltonian mapping – the analysis differs: using tools from the Schrieffer-Wolff transformation one can find precise bounds on intervals in which the energies in the low-energy sector must lie, gaining fine control over the relation between the acceptance probabilities of the circuit and the low-energy sector of the Hamiltonian. The main lemma we use from [23] is adopted from the proof of Lemma 26 in their work.

▶ **Lemma 15** (Small-penalty clock construction, adopted from Lemma 26 in [23]). *Let U_n be a quantum verification circuit for inputs x, $|x| = n$, where U_n consists of $T = \text{poly}(n)$ gates from some universal gate-set using at most 2-local gates. Denote $P(\psi)$ for the probability that U_n accepts $(x, |\psi\rangle)$, and let H^x_{FK} be the corresponding 3-local Hamiltonian from the circuit-to-Hamiltonian mapping in [34] with a ϵ-factor in front of H_{out}, as in Eq. (2). Then for all $\epsilon \leq c/T^3$ for some constant $c > 0$, we have that low-energy subspace $\mathcal{S}_\epsilon$ of H, i.e.*

$$S_\epsilon = span\{|\Phi\rangle : \langle\Phi| H |\Phi\rangle \leq \epsilon\}$$

has that its eigenvalues λ_i satisfy

$$\lambda_i \in \left[\epsilon\frac{1-P(\psi_i)}{T+1} - \mathcal{O}(T^3\epsilon^2), \epsilon\frac{1-P(\psi_i)}{T+1} + \mathcal{O}(T^3\epsilon^2)\right], \tag{5}$$

where $\{|\psi_i\rangle\}$ are the eigenstates of the Mariott-Watrous operator of the circuit U_n given by Eq. (1).

Having a QCMA-verifier with the CNOT-trick of Lemma 14 ensures that in Lemma 15 all $|\psi_i\rangle$ are computational basis states, as the CNOT-trick diagonalizes the Mariott-Watrous operator. The small-penalty clock construction, in combination with the CNOT-trick and some properties of the class QCMA, allows us to show QCMA-hardness of guidable local Hamiltonian problems in a wide range of parameter settings.

▶ **Theorem 16.** CGaLH(k, δ, ζ) *is* QCMA*-hard under randomized reductions for $k \geq 2$, $\zeta \in (1/\text{poly}(n), 1 - 1/\text{poly}(n))$ and $\delta = 1/\text{poly}(n)$.*

Proof. We will only state a "basic version" reduction, which uses basis states as guiding states which trivially satisfy the conditions of Definition 1, for which we prove completeness and soundness. One can improve its parameters in terms of the achievable fidelity and locality domains, which is done in the full manuscript [48].

The basic reduction. Let $\langle U_n, p_1, p_2\rangle$ be a QCMA promise problem. By the result of [6], there exists randomized reduction to a UQCMA (which is QCMA but with a unique accepting witness in the YES-case) promise problem $\langle \hat{U_n}, \hat{p}_1, \hat{p}_2\rangle$, $\hat{p}_1 - \hat{p}_2 \geq 1/q(n)$ for some polynomial q, which uses witnesses $y \in \{0,1\}^{p(n)}$ for some polynomial $p(n)$ and uses at most $T = \text{poly}(n)$ gates. We will now apply the following modifications to the UQCMA instance:

1. First, we force the witness to be classical by adding another register to which we "copy" all bits of y (through CNOT operations), before running the actual verification protocol – i.e. we use the CNOT trick of Lemma 14, which diagonalizes the corresponding Marriot-Watrous operator in the computational basis.

2. We apply *error reduction* to the circuit. This is done by applying the so-called "Marriot and Watrous trick" for error reduction, described in [39], which allows one to repeat the verification circuit several times whilst re-using the same witness. It is shown in [39], Theorem 3.3, that for any quantum circuit V_n using $T = \text{poly}(n)$ 2-qubit gates which decides on acceptance or rejectance of an input x, $|x| = n$, using a $p(n)$-qubit witness $|\psi\rangle$ for some polynomial p, satisfying completeness and soundness probabilities c, s such that $c - s \geq 1/q(n)$ there is another circuit $\tilde{V}_n$ that again uses a $p(n)$-qubit witness $|\psi\rangle$ but has completeness and soundness $1 - 2^{-r}$ and 2^{-r}, respectively, at the cost of using $\tilde{T} = \mathcal{O}(q^2 rT)$ gates.

Let the resulting protocol be denoted by $\langle \tilde{U}_n, \tilde{c}, \tilde{s}\rangle$, where $\hat{U}_n$ has an input register A, a witness register W and ancilla register B, uses $\tilde{T} = \mathcal{O}(q^2 rT)$ gates and has completeness and soundness $C = 1 - 2^{-r}$ and $\hat{s} = 2^{-r}$. We denote y^* for the (unique) witness with acceptance probability $\geq C$ in the YES-case. We keep r as a parameter to be tuned later in our construction. We will also write $P(y) := \Pr[\hat{U} \text{ accepts } (y)]$. Now consider the 4-local Hamiltonian

$$H^x = H_{yes} \otimes |0\rangle\langle 0|_D + H_{yes} \otimes |1\rangle\langle 1|_D\,, \tag{6}$$

where $H_{yes} = H^x_{\text{FK}}$ is the Hamiltonian given by Eq. (2) using the circuit $\hat{U}_n$ and parameter ϵ and H_{no} is given by

$$H_{no} = \sum_{i=0}^{R-1} |1\rangle\langle 1|_i + bI, \tag{7}$$

where R is the total size of the registers A, W, B and the clock register C, and $b > 0$ is yet another tunable parameter. Note that H_{no} has a *unique* ground state with energy b given by the all zeros state, and the spectrum after that increases in steps of 1 (and so it in particular has a *spectral gap* of 1). We also have that $\|H_{no}\| = R + b = \text{poly}(n)$. As a guiding state in the YES-case we will use the following basis state

$$|u_{\text{yes}}\rangle = |x\rangle_A |y^*\rangle_W |0\ldots0\rangle_B |0\rangle_C |0\rangle_D\,, \tag{8}$$

which satisfies $(\langle\eta(y^*)| \langle 0|_D)\,|u_{\text{yes}}\rangle = 1/\sqrt{(T+1)} = \mathcal{O}(1/\text{poly}(N))$, with $|\eta(y^*)\rangle$ being the history state of witness y^* for Hamiltonian H_{yes}. In the NO-case, we will show that the state

$$|u_{\text{no}}\rangle = |0\ldots0\rangle_{AWBC} |1\rangle_D\,, \tag{9}$$

will be in fact the ground state. We will now show that setting $b := \mathcal{O}(1/\tilde{T}^7)$ and $\epsilon := \mathcal{O}(1/\tilde{T}^5)$, our reduction achieves the desired result.

Completeness. Let us first analyse the YES-case. By Lemma 15, we have that the eigenvalue $\lambda(y)$ corresponding to the witness y^* is upper bounded by

$$\lambda(y^*) \leq \epsilon\frac{2^{-r}}{\tilde{T}+1} + \mathcal{O}(\tilde{T}^3\epsilon^2).$$

On the other hand, we have that for any $y \neq y^*$

$$\lambda(y) \geq \epsilon\frac{1-2^{-r}}{\tilde{T}+1} - \mathcal{O}(\tilde{T}^3\epsilon^2) = \Omega\left(\frac{1}{\tilde{T}^6}\right)$$

for our choice of ϵ and $r \geq 1$. Hence, for our choice of ϵ we must have that the ground state $|\psi\rangle$ of H_{yes} is unique and has a spectral gap that can be bounded as

$$\gamma(H_{yes}) \geq \epsilon \frac{1-2^{-r+1}}{\tilde{T}+1} - \mathcal{O}\left(\tilde{T}^3\epsilon^2\right) = \Omega\left(\frac{1}{\tilde{T}^6}\right), \tag{10}$$

for some $r \geq \Omega(1)$ (we will pick r to be much larger later). Let us consider the fidelity of the history state $|\eta(y^*)\rangle$ with the actual ground state. First, we have that the energy of $|\eta(y^*)\rangle$ is upper bounded by

$$\langle\eta(y^*)|\, H_{yes}\, |\eta(y^*)\rangle \leq \epsilon \frac{2^{-r}}{\tilde{T}+1} = \mathcal{O}\left(\frac{2^{-r}}{\tilde{T}^6}\right),$$

which follows directly from Eq. 4 and the fact that $P(y^*) \geq 1 - 2^{-r}$. We can write $|\eta(y^*)\rangle$ in the eigenbasis of H_{yes} as $|\eta(y^*)\rangle = \alpha\,|\psi\rangle + \sqrt{1-\alpha^2}|\psi^\perp\rangle$, for some real number $\alpha \in [0,1]$, where $|\psi\rangle$ is the actual ground state of H_{yes} and $|\psi^\perp\rangle$ another state orthogonal to $|\psi\rangle$. We have that the energy of $|\eta(y^*)\rangle$ is upper bounded by

$$\langle\eta(y^*)|\, H_{yes}\, |\eta(y^*)\rangle \leq \epsilon \frac{2^{-r}}{\tilde{T}+1} = \mathcal{O}\left(\frac{2^{-r}}{\tilde{T}^6}\right).$$

On the other hand, the energy of $|\eta(y^*)\rangle$ is lower bounded by

$$\langle\eta(y^*)|\, H_{yes}\, |\eta(y^*)\rangle = \alpha^2 \langle\psi|\, H_{yes}\, |\psi\rangle + (1-\alpha^2)\langle\psi^\perp|H_{yes}|\psi^\perp\rangle \geq \Omega\left(\frac{1-\alpha^2}{\tilde{T}^6}\right),$$

using the fact that H_{yes} is PSD. Combining the upper and lower bounds, we find

$$\alpha^2 = |\,\langle\eta(y^*)|\psi\rangle\,|^2 \geq 1 - \mathcal{O}\left(2^{-r}\right), \tag{11}$$

which can be made $\geq 1 - 2^{-c\tilde{T}}$ for some $r = c\tilde{T} + \mathcal{O}(1)$. Hence, we have that the fidelity of $|u_{\text{yes}}\rangle$ with the unique ground state of H can be lower bounded as

$$\begin{aligned}|\langle u_{\text{yes}}|\psi\rangle|^2 &\geq 1 - \left(\sqrt{1 - |\,\langle u_{\text{yes}}|\,(|\eta(y^*)\rangle\,|0\rangle)|^2} + \sqrt{1 - |(\langle\eta(y^*)|\,\langle 0|)\,|\psi\rangle\,|^2}\right)^2 \\ &\geq 1 - \left(\sqrt{1 - \frac{1}{\tilde{T}+1}} + 2^{-c\tilde{T}/2}\right)^2 \\ &\geq \Omega\left(\frac{1}{\tilde{T}}\right),\end{aligned}$$

as desired.

Soundness. We have that all witnesses y get accepted by $\hat{U}$ with at most an exponentially small probability, and hence have that $H_{yes} \succeq \Omega(1/\tilde{T}^6)$. By our choice b we have therefore ensured that the ground state in the NO-case must be the state given by Eq. (9), which has energy $b = \Omega(1/\tilde{T}^7)$. Hence, the promise gap between YES and NO cases is $\delta = \Omega(1/\tilde{T}^7) = \Omega(1/q^2T^8) = 1/\text{poly}(n)$.

In the full version [48] the rest of the proof can be found, which uses similar tricks as in [17, 18] to improve the basic construction in terms of the fidelity range and locality. ◀

Now that we have established QCMA-completeness for CGaLH*, we get QCMA-completeness for QGaLH *for free* for the same range of parameter settings, as the latter is a generalization of the former (containing CGaLH* as a special case), and containment holds by the same argument as used in the proof of Theorem 2 in [18]. However, with just a little bit more work we can see that QCMA-hardness for QGaLH actually persists even when the overlap is *exponentially* close to one. A proof of this is given in the full version [48].

3.2 Spectral amplification

In this subsection we will discuss spectral amplification, which is the key technique behind showing the containment results of Theorem 6. Let $H = \sum_{i=0}^{m-1} H_i$ be a Hamiltonian on n qubits which is a sum of k-local terms H_i, which satisfies $\|H\| \leq 1$. Since H is Hermitian, we can write H as

$$H = \sum_{i=0}^{2^n-1} \lambda_i \left|\psi_i\right\rangle \left\langle\psi_i\right|,$$

where $\lambda_i \in [-1, 1]$ (by assumption on the operator norm) denotes the i'th eigenvalue of H with corresponding eigenvector $|\psi_i\rangle$. Consider a polynomial $P \in \mathbb{R}[x]$ of degree d, and write

$$P(x) = a_0 + a_1 x + \cdots + a_d x^d.$$

The *polynomial spectral amplification* of H for P is then defined as

$$\begin{aligned} P(H) &= a_0 I + a_1 H + \cdots + a_d H^d \\ &= a_0 I + a_1 \sum_{i=0}^{2^n-1} \lambda_i \left|\psi_i\right\rangle \left\langle\psi_i\right| + \cdots + a_d \sum_{i=0}^{2^n-1} \lambda_i^d \left|\psi_i\right\rangle \left\langle\psi_i\right| \\ &= \sum_{i=0}^{2^n-1} P(\lambda_i) \left|\psi_i\right\rangle \left\langle\psi_i\right|. \end{aligned}$$

Now for $\alpha \in [-1, 1]$, denote

$$\Pi_\alpha = \sum_{\{i:\lambda_i \leq \alpha\}} \left|\psi_i\right\rangle \left\langle\psi_i\right| \tag{12}$$

for the projection on all eigenstates of H which have eigenvalues at most α, which we will call a *low-energy projector* of H. Note that for any $\alpha \geq \lambda_0$, we must have that $\Pi_{\mathrm{gs}}\Pi_\alpha = \Pi_\alpha\Pi_{\mathrm{gs}} = \Pi_{\mathrm{gs}}$. We can utilize such a projector to solve $\mathsf{CGaLH}(k, \delta, \zeta)$, simply by computing $\|\Pi_\alpha |u\rangle\|$ for $\alpha = a$ given a classically evaluatable state $|u\rangle$. To see why this works, note that in the YES-case, for the witness desc(u) we have that $\|\Pi_a |u\rangle\| \geq \|\Pi_{\mathrm{gs}} |u\rangle\| \geq \sqrt{\zeta}$ and in the NO-case we have that $\|\Pi_a |v\rangle\| = 0$ for all states, which means that the two cases are separated by $\sqrt{\zeta}$. However, it is unlikely that an efficient description exists of Π_a, and even if it did, it would not be k-local and therefore $\|\Pi_a |u\rangle\|$ would not even be necessarily efficiently computable.

The idea is now to approximate this low-energy projector Π_α by a polynomial in H. To see this, note that Π_α can be written exactly as

$$\Pi_\alpha = \frac{1}{2}\left(1 - \mathrm{sgn}(H - \alpha I)\right),$$

where sgn(x) is the sign function, which for our purposes is defined on $\mathbb{R} :\to \mathbb{R}$ as

$$\mathrm{sgn}(x) = \begin{cases} 1 & \text{if } x > 0, \\ -1 & \text{if } x \leq 0. \end{cases}$$

From [32] we can then use the polynomial approximation of the sign function, which can subsequently be shifted to obtain the desired approximate low-energy projector $\tilde{\Pi}_a$.

▶ **Lemma 17** (Polynomial approximation to the sign function, from [32]). *For all $\delta' > 0, \epsilon' \in (0, 1/2)$ there exists an efficiently computable odd polynomial $P \in \mathbb{R}[x]$ of degree $d = \mathcal{O}\left(\frac{\log(1/\epsilon')}{\delta'}\right)$, such that*

- *for all $x \in [-2, 2] : |P(x)| \leq 1$, and*
- *for all $x \in [-2, 2] \setminus (-\delta', \delta') : |P(x) - sgn(x)| \leq \epsilon'$.*

Since Lemma 17 holds on the entire interval $[-2, 2]$, choosing any $\alpha \in [-1, 1]$ and scaling the $\text{sgn}(x)$ function with the factor $1/2$ will ensure that the error, as in the lemma, will be $\leq \epsilon/2$. Let $q_\alpha(x) : \mathbb{R} \to [0, 1]$ defined as $q_\alpha(x) = \frac{1}{2}(1 - \text{sgn}(x - \alpha))$ be this function, with polynomial approximation $Q_\alpha \in \mathbb{R}[x]$ of degree d. Note that Q_α can be written as a function of P as $Q_\alpha(x) = \frac{1}{2}(1 - P(x - \alpha))$. We will write $\tilde{\Pi}_\alpha = Q_\alpha(H)$ for the corresponding polynomial approximation of the approximate low-energy ground state "projector". Note that $\tilde{\Pi}_\alpha$ is Hermitian (since H is Hermitian), but that $\tilde{\Pi}_\alpha$ is no longer necessarily a projector and therefore $\tilde{\Pi}_\alpha^2 \neq \tilde{\Pi}_\alpha$. If we now replace Π_α in $\|\Pi_\alpha |u\rangle\|$ by $\tilde{\Pi}_\alpha$, we get $\left\|\tilde{\Pi}_\alpha |u\rangle\right\| = \sqrt{\langle u| \tilde{\Pi}_\alpha^\dagger \tilde{\Pi}_\alpha |u\rangle} = \sqrt{\langle u| \tilde{\Pi}_\alpha^2 |u\rangle} = \sqrt{\langle u| (Q_\alpha(H))^2 |u\rangle}$, which means that we have to evaluate up to degree $2d$ powers of H. The next lemma (proof in full version [48]) will give an upper bound on the number of expectation values that have to be computed when evaluating a polynomial of H of degree d.

▶ **Lemma 18.** *Given access to a classically evaluatable state $|u\rangle$, a Hamiltonian $H = \sum_{i=0}^{m-1} H_i$, where each H_i acts on at most k qubits non-trivially, and a polynomial $P[x]$ of degree d, there exists a classical algorithm that computes $\langle u| P(H) |u\rangle$ in $\mathcal{O}(m^d)$ computations of $\langle u| O_i |u\rangle$, where the observables $\{O_i\}$ are at most kd-local.*

All that remains to show is that for constant promise gap δ, using a good enough approximation $\tilde{\Pi}_\alpha$ with a suitable choice of α, will ensure that we can still distinguish the two cases in the $\mathsf{CGaLH}(k, \delta, \zeta)$ problem in a polynomial (resp. quasi-polynomial number of computations in m when $\zeta = \Omega(1)$ (resp. $\zeta = 1/\text{poly}(n)$).

▶ **Theorem 19.** *Let $H = \sum_{i=0}^{m-1} H_i$ be some Hamiltonian, and $desc(u)$ be a description of a classically evaluatable state $u \in \mathbb{C}^{2^n}$. Let $a, b \in [-1, 1]$ such that $b - a \geq \delta$, where $\delta > 0$ and let $\zeta \in (0, 1]$. Consider the following two cases of H, with the promise that either one holds:*

(i) *H has an eigenvalue $\leq a$, and $\|\Pi_{gs} |u\rangle\|^2 \geq \zeta$ holds, or*

(ii) *all eigenvalues of H are $\geq b$.*

Then there exists a classical algorithm that distinguishes between cases (i) and (ii) using

$$\mathcal{O}\left(m^{c\left(\log\left(1/\sqrt{\zeta}\right)\right)/\delta)}\right)$$

computations of local expectation values, for some constant $c > 0$.

Proof. Let $\tilde{\Pi}_\alpha := Q_\alpha(H)$, where Q is a polynomial of degree d, be the approximate low-energy projector that approximates $\Pi_\alpha = \frac{1}{2}(1 - \text{sgn}(H - (\alpha I)))$. We set $\alpha := \frac{a+b}{2}$, $\delta' := \delta/2$ and $\epsilon' = 1/10$. We propose the following algorithm:

1. Compute $\left\|\tilde{\Pi}_a |u\rangle\right\|$ using a polynomial of degree $2d$ where $d = \mathcal{O}(\log(1/\epsilon'))/\delta'$, for $\epsilon' := \frac{1}{10}\sqrt{\zeta}$ and $\delta' = \delta/2$.
2. If $\left\|\tilde{\Pi}_\alpha |u\rangle\right\| \geq \frac{9}{10}\sqrt{\zeta}$ output (i), and otherwise output (ii).

Clearly, by Lemma 18, we have that this can be done in at most $\mathcal{O}\left(m^{c\left(\log\left(1/\sqrt{\zeta}\right)\right)/\delta)}\right)$ computations of expectation values of local observables, for some constant c. Let us now prove the correctness of the algorithm. Note that we can write $\tilde{\Pi}_\alpha$ as

$$\tilde{\Pi}_\alpha = \sum_{i=0}^{2^n-1} Q(\lambda_i) |\psi_i\rangle \langle\psi_i| ,$$

where we have that

$$\begin{cases} 1-\sqrt{\zeta}/2 \leq Q(\lambda_i) \leq 1 & \text{if } \lambda_i \leq a, \\ 0 \leq Q(\lambda_i) \leq \zeta/2 & \text{if } \lambda_i \geq b, \\ 0 \leq Q(\lambda_i) \leq 1 & \text{else,} \end{cases}$$

by Lemma 17. Let us analyse both cases (i) and (ii) separately.

(i) H has an eigenvalue $\leq a$, and $\|\Pi_{\text{gs}} |u\rangle\|^2 \geq \zeta$ holds.

$$\begin{aligned} \left\|\tilde{\Pi}_\alpha |u\rangle\right\| &\geq \left\|\tilde{\Pi}_\alpha \Pi_{\text{gs}} |u\rangle\right\| \\ &= \left\|\Pi_\alpha \Pi_{\text{gs}} |u\rangle - (\Pi_\alpha - \tilde{\Pi}_\alpha)\Pi_{\text{gs}} |u\rangle\right\| \\ &= \left\|\Pi_{\text{gs}} |u\rangle - \left(\sum_{i:\lambda_i\leq\alpha} (1-Q(\lambda_i)) |\psi_i\rangle\langle\psi_i| - \sum_{i:\lambda_i>\alpha} Q(\lambda_i) |\psi_i\rangle\langle\psi_i|\right) \Pi_{\text{gs}} |u\rangle\right\| \\ &\geq \left\|\Pi_{\text{gs}} |u\rangle - \left(\sum_{i:\lambda_i\leq\alpha} \frac{1}{10} |\psi_i\rangle\langle\psi_i|\right) \Pi_{\text{gs}} |u\rangle\right\| \\ &= \left\|\Pi_{\text{gs}} |u\rangle - \frac{1}{10}\Pi_\alpha \Pi_{\text{gs}} |u\rangle\right\| \\ &= (1-\frac{1}{10})\|\Pi_{\text{gs}} |u\rangle\| \\ &\geq \frac{9}{10}\sqrt{\zeta}. \end{aligned}$$

(ii) All eigenvalues of H are $\geq b$. We must have that $\left\|\tilde{\Pi}_\alpha |u\rangle\right\| \leq \frac{1}{2}\sqrt{\zeta}$, since $\lambda_i \geq b$ for all $i \in \{0, \ldots, 2^n-1\}$.

Hence, we have that the promise gap between both cases is lower bounded by $\frac{9}{10}\sqrt{\zeta} - \frac{1}{2}\sqrt{\zeta} = \frac{2}{5}\sqrt{\zeta}$, which is $1/\text{poly}(n)$ when $\zeta \geq 1/\text{poly}(n)$. ◀

▶ Remark 20. It should be straightforward to adopt the same derivation as above to a more general setting by considering sparse matrices, a promise with respect to the fidelity with the low-energy subspace (i.e. all states with energy $\leq \lambda_0 + \gamma$ for some small γ), as well as $\epsilon > 0$ for ϵ-classically evaluatable states.

3.3 Upper bound on QCPCP with constant proof queries

Here we show that QCPCP with a constant number of proof queries is contained in $\mathsf{BQP}^{\mathsf{NP}[1]}$, i.e. in BQP with only a single query to an NP-oracle. The full proof is rather long, but the idea is simple: just as is the case for QPCP, a *quantum* reduction can be used to transform a QCPCP system into a local Hamiltonian problem. However, since the proof is now classical, one can directly learn a diagonal (i.e. classical) Hamiltonian that captures the input/output behaviour of the QCPCP-circuit on basis state inputs. The main technical work required is to derive sufficient parameters in the reduction, thereby ensuring that the reduction succeeds with the desired success probability.

We will use the following two lemmas, whose proofs can be bound in the full version [48].

▶ **Lemma 21.** *Let $H = \sum_{i\in[m]} w_i H_i$ be a k-local Hamiltonian consisting of weights $w_i \in [0,1]$ such that $\sum_{i\in[m]} w_i = W$, and k-local terms H_i for which $\|H_i\| \leq 1$ for all $i \in [m]$. Let $\Omega_{\geq\gamma} = \{i | w_i \geq \gamma\}$ and $\Omega_{<\gamma} = [m] \setminus \Omega_{\geq\gamma}$, for some parameter $\gamma \in [0,1]$. Suppose $\tilde{H} = \sum_{i\in\Omega_{\geq\gamma}} \tilde{w}_i \tilde{H}_i$ is another Hamiltonian such that, for all $i \in \Omega_{\geq\gamma}$, we have $|\tilde{w}_i - w_i| \leq \epsilon_0$ and $\left\|H_i - \tilde{H}_i\right\| \leq \epsilon_1$. Then*

$$\left\|H - \tilde{H}\right\| \leq m(\gamma + \epsilon_0) + (W + m\epsilon_0)\epsilon_1$$

▶ **Lemma 22** (Upper bound on the non-uniform double dixie cup problem)**.** *Given samples from the set $N = [n]$, according to a distribution $\mathcal{P}$, consider the subset $M_\gamma \subseteq N$ such that $M_\gamma = \{i \in N : \mathcal{P}(i) \geq \gamma\}$, for some $\gamma \in [0,1]$. Let $T_m^{\mathcal{P}}(M)$ be the random variable indicating the first time that all elements in M_γ have been sampled at least m times when sampling from N over the distribution $\mathcal{P}$. Write $T_m(S)$ when the distribution over some set S is uniform. Then we have that*

$$\mathbb{E}[T_m^{\mathcal{P}}(M_\gamma)] \leq \mathbb{E}[T_m(\lceil 1/\gamma \rceil)],$$

where $\mathbb{E}[T_m(\lceil 1/\gamma\rceil)] = \lceil 1/\gamma\rceil \ln\lceil 1/\gamma\rceil + (m-1)\lceil 1/\gamma\rceil \ln\ln\lceil 1/\gamma\rceil + \mathcal{O}\left(\lceil 1/\gamma\rceil\right).$

Let us now consider the quantum algorithm used to learn the diagonal Hamiltonian whose spectrum encodes the acceptance probabilities of the QCPCP-verifier. Let V_x be the QCPCP-verifier circuit with the input x hardcoded into it. The idea of the algorithm is that it runs V_x many times, simultaneously gathering statistics on which indices are most likely to be queried by V_x (which is independent of the proof when the verifier is non-adaptive) as well as the probability of acceptance given that the proof locally looks like a string $z \in \{0,1\}^q$. For every run, indexed by $t \in [T]$ for some $T \in \mathbb{N}$, this generates a tuple $O^{t,z} = ((i_1^{t,z}, \dots, i_k^{t,z}), o^{t,z})$, in which the proof y was supposed to be queried at indices $i_1, \dots, i_q$, and in which those bits were assigned the values $y_{i_1} = z_1, \dots, y_{i_q} = z_q$, and where o is the accept/reject measurement outcome. It repeats this process T times for every z. The resulting algorithm can be specified as follows:

1. For $z \in \{0,1\}^q$:
   a. Run V_x for a total of T times to obtain samples $\{O^{t,z}\}_{t\in[T]}$.
   b. For all observed $(i_1^{t,z}, \dots, i_q^{t,z})$, set
$$\tilde{\lambda}_{x,(i_1,\dots,i_q)}(z) := \frac{\#\text{ times } o^{t,z} = 1 \text{ and } i_1,\dots,i_q \text{ observed}}{\#\text{ times } i_1,\dots,i_q \text{ observed}}.$$
2. Set
$$\tilde{\mathcal{P}}_x(i_1,\dots,i_q) = \sum_{z\in\{0,1\}^q} \frac{\#\text{ times } (i_1^{t,z},\dots,i_q^{t,z}) \text{ observed}}{2^q T},$$
3. For any estimated $\tilde{\mathcal{P}}_x(i_1,\dots,i_q) \leq \gamma$ remove both $\tilde{\mathcal{P}}_x(i_1,\dots,i_q)$ and associated $\tilde{\lambda}_{x,(i_1,\dots,i_q)}(z)$ for all z, and output all of the remaining ones.

The resulting diagonal Hamiltonian will then be constructed as

$$\tilde{H}_x = \sum_{(i_1,\dots,i_q)\in\Omega_{\geq\gamma}} \tilde{\mathcal{P}}_x(i_1,\dots,i_q)\tilde{H}_{x,(i_1,\dots,i_q)},$$

where

$$\tilde{H}_{x,(i_1,\dots,i_q)} = \sum_{z\in\{0,1\}^q} (1 - \tilde{\lambda}_{x,(i_1,\dots,i_q)})(z)|z\rangle\langle z|_{i_1,\dots,i_q}.$$

By Lemma 21 we can upper bound the difference between the Hamiltonian and the learned Hamiltonian. The next lemma shows that all relevant parameters can be learned up to inverse polynomial precision in polynomial time.

▶ **Lemma 23.** *Let $q \in \mathbb{N}$ be some constant and x an input with $|x| = n$. Consider a* $\mathsf{QCPCP}[q]$ *protocol with verification circuit V_x (which is V but with the input x hardcoded into the circuit), and proof $y \in \{0,1\}^{p(n)}$, and let*

$$\mathcal{P}_x(i_1, \ldots, i_q) = \mathbb{P}\left[V_x \textit{ queries the proof at indices } (i_1, \ldots, i_q)\right]$$

and

$$\lambda_{x,(i_1,\ldots,i_q)}(z) = \mathbb{P}\begin{bmatrix} V_x \textit{ accepts given proof bits } i_1, \ldots, i_q \textit{ are queried} \\ \textit{and are given by } y_{i_1} = z_1, \ldots, y_{i_q} = z_q \end{bmatrix}.$$

Let $\Omega = \{(i_1, \ldots, i_q) : i_j \in [p(n)], \forall j \in [q]\}$, $\Omega_{\geq\gamma} = \{(i_1, \ldots, i_q) \in \Omega | \mathcal{P}_x(i_1, \ldots, i_q) \geq \gamma\}$ and $\Omega_{<\gamma} = \Omega \setminus \Omega_{\geq\gamma}$, for some parameter $\gamma \in [0,1]$. Then there exists a quantum algorithm that, for all $(i_1, \ldots, i_q) \in \Omega_{\geq\gamma}$ and all $z \in \{0,1\}^q$, provides estimates $\tilde{\mathcal{P}}_x(i_1, \ldots, i_q)$ and $\tilde{\lambda}_{x,(i_1,\ldots,i_q)}(z)$ such that

$$\left|\tilde{\mathcal{P}}_x(i_1, \ldots, i_q) - \mathcal{P}_x(i_1, \ldots, i_q)\right| \leq \epsilon_0,$$

and

$$\left|\tilde{\lambda}_{x,(i_1,\ldots,i_q)}(z) - \lambda_{x,(i_1,\ldots,i_q)}(z)\right| \leq \epsilon_1,$$

with probability $1 - \delta$, and runs in time $\mathrm{poly}(n, 1/\gamma, 1/\epsilon_0, 1/\epsilon_1, 1/\delta)$.

Proof. Let us now show that there exists a T not too large such that the criteria of the theorem are satisfied. Since the $\mathcal{P}_x(i_1, \ldots, i_q)$ form a discrete distribution over the set Ω, where $|\Omega| = \binom{n}{q} \leq \left(\frac{en}{q}\right)^q$ which for constant q is polynomial in n, we know by a standard result in learning theory (see for example [19]) that a total of

$$\Theta\left(\frac{|\Omega| + \log(1/\delta_0)}{\epsilon_0^2}\right)$$

samples of $O^{t,z}$ (the "z"-value is in fact irrelevant here) suffices to get, with probability at least $1 - \delta_0$, estimates $\tilde{\mathcal{P}}_x(i_1, \ldots, i_q)$ which satisfy

$$\left|\tilde{\mathcal{P}}_x(i_1, \ldots, i_q) - \mathcal{P}_x(i_1, \ldots, i_q)\right| \leq \epsilon_0.$$

To learn estimates $\tilde{\lambda}_{x,(i_1,\ldots,i_q)}(z)$ for a single index configuration $(i_1, \ldots, i_q)$ and proof configuration z, Hoeffding's inequality tells us that we only need

$$\mathcal{O}\left(\frac{\log(1/\delta_1)}{\epsilon_1^2}\right)$$

samples of $O^{t,z}$ to have that $\left|\tilde{\lambda}_{x,(i_1,\ldots,i_q)}(z) - \lambda_{x,(i_1,\ldots,i_q)}(z)\right| \leq \epsilon_1$, with probability $1 - \delta_1$. This means that any index configuration $(i_1, \ldots, i_q)$ such that $\mathcal{P}_x(i_1, \ldots, i_q) \geq \gamma$ needs to appear $\mathcal{O}\left(\frac{\log(1/\delta_1)}{\epsilon_1^2}\right)$ many times, to get a good estimate of $\tilde{\lambda}_{x,(i_1,\ldots,i_q)}(z)$. Lemma 22 shows that the expected number of samples needed such that this condition is met is upper bounded by

$$\left\lceil\frac{1}{\gamma}\right\rceil \ln\left\lceil\frac{1}{\gamma}\right\rceil + \left(\mathcal{O}\left(\frac{\log(1/\delta_1)}{\epsilon_1^2}\right) - 1\right)\left\lceil\frac{1}{\gamma}\right\rceil \ln\ln\left\lceil\frac{1}{\gamma}\right\rceil + \mathcal{O}\left(\left\lceil\frac{1}{\gamma}\right\rceil\right),$$

which by Markov's inequality means that

$$\frac{1}{\delta_\lambda}\left(\left\lceil\frac{1}{\gamma}\right\rceil \ln\left\lceil\frac{1}{\gamma}\right\rceil + \left(\mathcal{O}\left(\frac{\log\left(1/\delta_1\right)}{\epsilon_1^2}\right) - 1\right)\left\lceil\frac{1}{\gamma}\right\rceil \ln\ln\left\lceil\frac{1}{\gamma}\right\rceil + \mathcal{O}\left(\left\lceil\frac{1}{\gamma}\right\rceil\right)\right)$$

samples of $O^{t,z}$ suffice to turn this into an algorithm that achieves success probability $\geq 1-\delta_\lambda$. To ensure that our entire algorithm succeeds with probability $1-\delta$, we require that

$$(1-\delta_\lambda)^{2^q}(1-\delta_0)(1-\delta_1)^{2^q\left\lceil\frac{1}{\gamma}\right\rceil} \geq 1-\delta,$$

which can be achieved by setting $\delta_\lambda = \delta/(2^{q+2})$, $\delta_0 = \delta/4$ and $\delta_1 = \delta/(\lceil 1/\gamma\rceil 2^{q+2})$. Both the statistics for probabilities over the set of indices, as well as the output probabilities, are gathered at the same time. This means that the requirements on the number of samples needed for both estimations can be met at the same time, therefore the total number of samples T that we must take satisfies

$$T \geq \max\{\Theta\left(\frac{\left\lceil\frac{1}{\gamma}\right\rceil + \log\left(\frac{1}{\delta}\right)}{\epsilon_0^2}\right), \frac{2^{2(q+1)}}{\delta}\left(\left\lceil\frac{1}{\gamma}\right\rceil \ln\left\lceil\frac{1}{\gamma}\right\rceil + \mathcal{O}\left(\frac{q\log\left(\left\lceil\frac{1}{\gamma}\right\rceil/\delta\right)}{\epsilon_1^2}\right)\left\lceil\frac{1}{\gamma}\right\rceil \ln\ln\left\lceil\frac{1}{\gamma}\right\rceil\right)\},$$

which yields a total runtime of $\mathcal{O}(\text{poly}(n, \lceil 1/\gamma\rceil, 1/\delta, 1/\epsilon_1, 1/\epsilon_0))$ when $q = \mathcal{O}(1)$. ◀

Lemma 23 can then be combined with Lemma 21 to show that a diagonal Hamiltonian whose spectrum encodes the acceptance probabilities of V_x can be learned in polynomial time with high probability.

▶ **Lemma 24.** *Let $q \in \mathbb{N}$ be some constant, then there exists a quantum algorithm that can reduce any problem solvable by a* QCPCP$[q]$ *protocol, without access to the proof y, to a diagonal Hamiltonian $\tilde{H}_x$ with the following properties:*

- $x \in P_{yes} \Rightarrow \exists y \in \{0,1\}^{p(n)} : \langle y|\, \tilde{H}_x\, |y\rangle \leq \frac{1}{3} + \epsilon$
- $x \in P_{no} \Rightarrow \forall y \in \{0,1\}^{p(n)} : \langle y|\, \tilde{H}_x\, |y\rangle \geq \frac{2}{3} - \epsilon$.

This reduction succeeds with probability $1-\delta$ and runs in time poly$(n, 1/\epsilon, 1/\delta)$.

Finally, from Lemma 24 the main theorem follows, as a BQP verifier can perform the quantum reduction and, conditioned on succeeding, solve the resulting diagonal local Hamiltonian problem making only a single query to the NP-oracle.

▶ **Theorem 25.** *For all constant $q \in \mathbb{N}$, we have that* QCPCP$[q] \subseteq$ BQP$^{\mathsf{NP}[1]}$.

The full proofs of Theorem 25 and Lemma 24 are given in the full version [48].

References

1 Scott Aaronson. Computational complexity: Why quantum chemistry is hard. *Nature Physics*, 5:707–708, October 2009. doi:10.1038/nphys1415.

2 Scott Aaronson. Bqp and the polynomial hierarchy. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, STOC '10, pages 141–150, New York, NY, USA, 2010. Association for Computing Machinery. arXiv:0910.4698. doi:10.1145/1806689.1806711.

3 Scott Aaronson, DeVon Ingram, and William Kretschmer. The Acrobatics of BQP. In Shachar Lovett, editor, *37th Computational Complexity Conference (CCC 2022)*, volume 234 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:17, 2022. arXiv:2111.10409. doi:10.4230/LIPIcs.CCC.2022.20.

4 Dorit Aharonov, Itai Arad, Zeph Landau, and Umesh Vazirani. The detectability lemma and quantum gap amplification. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, pages 417–426, New York, NY, USA, 2009. arXiv:0811.3412. doi:10.1145/1536414.1536472.

5 Dorit Aharonov, Itai Arad, and Thomas Vidick. Guest column: the quantum pcp conjecture. *SIGACT News*, 44(2):47–79, June 2013. arXiv:1309.7495. doi:10.1145/2491533.2491549.

6 Dorit Aharonov, Michael Ben-Or, Fernando G.S.L. Brandão, and Or Sattath. The Pursuit of Uniqueness: Extending Valiant-Vazirani Theorem to the Probabilistic and Quantum Settings. *Quantum*, 6:668, March 2022. arXiv:0810.4840. doi:10.22331/q-2022-03-17-668.

7 Dorit Aharonov and Alex Bredariol Grilo. Stoquastic pcp vs. randomness. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1000–1023, 2019. arXiv:1901.05270. doi:10.1109/FOCS.2019.00065.

8 Tameem Albash and Daniel A. Lidar. Adiabatic quantum computation. *Rev. Mod. Phys.*, 90:015002, January 2018. arXiv:1611.04471 . doi:10.1103/RevModPhys.90.015002.

9 Itai Arad. A note about a partial no-go theorem for quantum pcp. *Quantum Info. Comput.*, 11(11–12):1019–1027, November 2011. arXiv:1012.3319.

10 Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, May 1998. doi:10.1145/278298.278306.

11 Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: a new characterization of np. *J. ACM*, 45(1):70–122, January 1998. doi:10.1145/273865.273901.

12 Bela Bauer, Sergey Bravyi, Mario Motta, and Garnet K. Chan. Quantum algorithms for quantum chemistry and quantum materials science. *Chemical Reviews*, 120(22):12685–12717, October 2020. arXiv:2001.03685. doi:10.1021/acs.chemrev.9b00829.

13 Paul Benioff. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of Statistical Physics*, 22(5):563–591, May 1980. doi:10.1007/BF01011339.

14 Jacob D. Biamonte, Jason Morton, and Jacob W. Turner. Tensor network contractions for #SAT. *Journal of Statistical Physics*, 160:1389–1404, June 2015. arXiv:1405.7375. doi:10.1007/s10955-015-1276-z.

15 Michael J Bremner, Richard Jozsa, and Dan J Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2126):459–472, 2011. arXiv:1005.1407. doi:10.1098/rspa.2010.0301.

16 Harry Buhrman, Jonas Helsen, and Jordi Weggemans. Quantum PCPs: on Adaptivity, Multiple Provers and Reductions to Local Hamiltonians. *CoRR*, March 2024. arXiv:2403.04841.

17 Chris Cade, Marten Folkertsma, Sevag Gharibian, Ryu Hayakawa, François Le Gall, Tomoyuki Morimae, and Jordi Weggemans. Improved Hardness Results for the Guided Local Hamiltonian Problem. In *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261, pages 32:1–32:19, 2023. arXiv:2207.10250. doi:10.4230/LIPIcs.ICALP.2023.32.

18 Chris Cade, Marten Folkertsma, and Jordi Weggemans. Complexity of the guided local Hamiltonian problem: Improved parameters and extension to excited states. *CoRR*, July 2022. arXiv:2207.10097.

19 Clément L Canonne. A short note on learning discrete distributions. *CoRR*, February 2020. arXiv:2002.11457.

20 Nai-Hui Chia, András Gilyén, Tongyang Li, Han-Hsuan Lin, Ewin Tang, and Chunhao Wang. Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, pages 387–400, New York, NY, USA, 2020. Association for Computing Machinery. arXiv:190.06151. doi:10.1145/3357713.3384314.

21 Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, STOC '71, pages 151–158, New York, NY, USA, 1971. Association for Computing Machinery. doi:10.1145/800157.805047.

22 Jordan Cotler, Hsin-Yuan Huang, and Jarrod R. McClean. Revisiting dequantization and quantum advantage in learning tasks. *CoRR*, December 2021. arXiv:2112.00811.

23 Abhinav Deshpande, Alexey V. Gorshkov, and Bill Fefferman. Importance of the spectral gap in estimating ground-state energies. *PRX Quantum*, 3:040327, December 2022. arXiv:2007.11582. doi:10.1103/PRXQuantum.3.040327.

24 Irit Dinur. The pcp theorem by gap amplification. *J. ACM*, 54(3):12–es, June 2007. doi:10.1145/1236457.1236459.

25 Andrew Drucker. A PCP characterization of AM. *ICALP*, pages 581–592, July 2011. arXiv:1002.3664. doi:10.1007/978-3-642-22006-7_49.

26 Richard P Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7), 1982. doi:10.1007/BF02650179.

27 Lance Fortnow. The role of relativization in complexity theory. *Bulletin of the EATCS*, 52:229–243, 1994. URL: https://bibbase.org/network/publication/fortnow-theroleofrelativizationincomplexitytheory-1994.

28 Sevag Gharibian and François Le Gall. Dequantizing the quantum singular value transformation: hardness and applications to quantum chemistry and the quantum pcp conjecture. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2022, pages 19–32, New York, NY, USA, 2022. Association for Computing Machinery. arXiv:2111.09079. doi:10.1145/3519935.3519991.

29 Sevag Gharibian and Justin Yirka. The complexity of simulating local measurements on quantum systems. *Quantum*, 3:189, September 2019. arXiv:1606.05626. doi:10.22331/q-2019-09-30-189.

30 Oded Goldreich. On promise problems: A survey. In *Theoretical Computer Science: Essays in Memory of Shimon Even*, pages 254–290. Springer, 2006. doi:10.1007/11685654_12.

31 Alex B. Grilo. *Quantum proofs, the local Hamiltonian problem and applications*. PhD thesis, Université Sorbonne Paris Cité, April 2018. URL: https://www.irif.fr/~abgrilo/thesis.pdf.

32 Guang Hao Low and Isaac L. Chuang. Hamiltonian simulation by uniform spectral amplification. *CoRR*, July 2017. arXiv:1707.05391.

33 Dhawal Jethwani, François Le Gall, and Sanjay K. Singh. Quantum-Inspired Classical Algorithms for Singular Value Transformation. In Javier Esparza and Daniel Král', editors, *45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020)*, volume 170 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 53:1–53:14, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. arXiv:1910.05699. doi:10.4230/LIPIcs.MFCS.2020.53.

34 Julia Kempe and Oded Regev. 3-local hamitonian is qma-complete. *Quantum Info. Comput.*, 3(3):258–264, May 2003. arXiv:0302079.

35 Alexei Y. Kitaev, Alexander Shen, and Mikhail N. Vyalyi. *Classical and quantum computation*. Number 47 in Graduate Studies in Mathematics. American Mathematical Society, 2002.

36 Leonid A. Levin. Universal sequential search problems. *Problemy peredachi informatsii*, 9(3):115–116, 1973.

37 Lin Lin and Yu Tong. Near-optimal ground state preparation. *Quantum*, 4:372, December 2020. arXiv:2002.12508. doi:10.22331/q-2020-12-14-372.

38 Hongbin Liu, Guang Hao Low, Damian S. Steiger, Thomas Häner, Markus Reiher, and Matthias Troyer. Prospects of quantum computing for molecular sciences. *Materials Theory*, 6(1):1–17, March 2022. arXiv:2102.10081. doi:10.1186/s41313-021-00039-z.

39 Chris Marriott and John Watrous. Quantum Arthur-Merlin games. *CCC*, pages 275–285, June 2004. arXiv:cs/0506068. doi:10.1007/s00037-005-0194-x.

40 Bryan O'Gorman, Sandy Irani, James Whitfield, and Bill Fefferman. Intractability of electronic structure in a fixed basis. *PRX Quantum*, 3:020322, May 2022. arXiv:2103.08215. doi:10.1103/PRXQuantum.3.020322.

41 David Poulin and Matthew B. Hastings. Markov entropy decomposition: A variational dual for quantum belief propagation. *Phys. Rev. Lett.*, 106:080403, February 2011. arXiv:1012.2050. doi:10.1103/PhysRevLett.106.080403.

42 Ran Raz and Avishay Tal. Oracle separation of bqp and ph. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, pages 13–23, New York, NY, USA, 2019. Association for Computing Machinery. doi:10.1145/3313276.3316315.

43 Norbert Schuch, Michael M. Wolf, Frank Verstraete, and J. Ignacio Cirac. Computational complexity of projected entangled pair states. *Phys. Rev. Lett.*, 98:140506, April 2007. arXiv:0611050. doi:10.1103/PhysRevLett.98.140506.

44 Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, pages 217–228, New York, NY, USA, 2019. Association for Computing Machinery. arXiv:1807.04271. doi:10.1145/3313276.3316310.

45 Barbara M. Terhal and David P. DiVincenzo. Adaptive quantum computation, constant depth quantum circuits and Arthur-Merlin games. *Quantum Information & Computation*, 4(2):134–145, March 2004. arXiv:quant-ph/0205133. doi:10.26421/QIC4.2-5.

46 Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, et al. The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*, 986:1–128, November 2022. arXiv:2111.05176. doi:10.1016/j.physrep.2022.08.003.

47 L G Valiant and V V Vazirani. Np is as easy as detecting unique solutions. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, pages 458–463, New York, NY, USA, 1985. Association for Computing Machinery. doi:10.1145/22145.22196.

48 Jordi Weggemans, Marten Folkertsma, and Chris Cade. Guidable local hamiltonian problems with implications to heuristic ansatze state preparation and the quantum pcp conjecture. *CoRR*, February 2023. arXiv:302.11578.

49 Pawel Wocjan, Dominik Janzing, and Thomas Beth. Two QCMA-complete problems. *Quantum Information & Computation*, 3(6):635–643, November 2003. arXiv:quant-ph/0305090.

50 Stathis Zachos. Probabilistic quantifiers and games. *Journal of Computer and System Sciences*, 36(3):433–451, 1988. doi:10.1016/0022-0000(88)90037-2.

A Direct Reduction from the Polynomial to the Adversary Method

Aleksandrs Belovs
Faculty of Computing, University of Latvia, Riga, Latvia

Abstract

The polynomial and the adversary methods are the two main tools for proving lower bounds on query complexity of quantum algorithms. Both methods have found a large number of applications, some problems more suitable for one method, some for the other.

It is known though that the adversary method, in its general negative-weighted version, is tight for bounded-error quantum algorithms, whereas the polynomial method is not. By the tightness of the former, for any polynomial lower bound, there ought to exist a corresponding adversary lower bound. However, direct reduction was not known.

In this paper, we give a simple and direct reduction from the polynomial method (in the form of a dual polynomial) to the adversary method. This shows that any lower bound in the form of a dual polynomial is actually an adversary lower bound of a specific form.

2012 ACM Subject Classification Theory of computation → Quantum query complexity

Keywords and phrases Polynomials, Quantum Adversary Bound

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.11

Funding *Aleksandrs Belovs*: This research is supported by the Latvian Quantum Initiative under European Union Recovery and Resilience Facility project no. 2.3.1.1.i.0/1/22/I/CFLA/001. Part of this work has been supported by the ERDF project number 1.1.1.5/18/A/020 "Quantum algorithms: from complexity theory to experiment."

Acknowledgements The author is thankful to Shalev Ben-David for the suggestion to apply this construction to the polynomial method, and to anonymous referees for their helpful comments on the previous versions of the paper.

1 Introduction

Proving lower bounds on quantum query complexity is a task that has attained significant attention. The reason is that it is essentially the only known way to prove limitations on the power of quantum algorithms. For instance, Bennett, Bernstein, Brassard, and Vazirani [14] proved a quantum query lower bound for the OR function using what later became known as the hybrid method. This demonstrates that there is no way to attain a better than Grover's [21] quadratic speed-up for an NP-search problem if we treat the latter as a black-box (an oracle). Powerful tools for proving quantum query lower bounds have been developed consequently: the polynomial method, and the adversary method, both in its original (positive-weighted) and improved (negative-weighted) formulations.

The polynomial method is due to Beals, Buhrman, Cleve, Mosca, and de Wolf [9], and it was inspired by a similar method used by Nisan and Szegedy [27, 28] to prove lower bounds on randomized query complexity. The method builds on the following observation: if $\mathcal{A}$ is a T-query quantum algorithm, then its acceptance probability on input x can be expressed as a degree-$2T$ multivariate polynomial in the input variables x_i. Beals et al. [9] used this method to re-prove the lower bound for the OR function from [14], and establish other results like a tight lower bound for all total symmetric Boolean functions. A landmark result obtained by this method is the lower bound for the collision problem by Aaronson and Shi [3]. Similarly

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 11; pp. 11:1–11:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

as Bennett et al.'s result [14], it shows that a black-box approach to finding a collision in a hash function by a quantum computer is doomed as well. This method has been popular ever after.

The original adversary method is due to Ambainis [4], and it is an improvement on the aforementioned hybrid method. The bound was strengthened by Ambainis himself [5] and Zhang [35] shortly afterwards. One of the appealing features of this method is its convenient combinatorial formulation, which resulted in a number of applications [7, 20, 15, 19]. However, the original formulation of the adversary bound was subject to several important limitations [35].

Partly in order to overcome these limitations, Høyer, Lee, and Špalek generalised the adversary bound in [22]. Departing from the semidefinite formulation of the original adversary bound by Barnum, Saks, and Szegedy [8], Høyer et al. showed that the same expression still yields a lower bound if one replaces non-negative entries by arbitrary real numbers. This *negative-weighted* formulation of the bound is strictly more powerful than the positive-weighted one, but it lacks the combinatorial convenience of the latter. The bound turned out to be useful for composed functions [22] and sum-problems [13, 12]. In a series of papers [31, 29, 30], Reichardt et al. surprisingly proved that the negative-weighted version of the bound is tight for bounded-error algorithms!

The polynomial method, on the other hand, is known to be non-tight. Ambainis [5] constructed a first super-linear separation between the two for total Boolean functions. This was later improved to an almost quartic separation by Aaronson, Ben-David, and Kothari [1], which is essentially tight [2]. For partial functions, the separations can be even more impressive [6].

The history of relationship between the adversary and the polynomial methods is rather interesting. For instance, the AND of ORs function allows for a very simple adversary lower bound [4], but its polynomial lower bound is more complicated and was only obtained more than a decade later. It was achieved independently by Sherstov [33], and Bun and Thaler [17] using the technique of dual polynomials [32]. The latter is the dual of an approximating polynomial in the sense of linear programming. Therefore, by strong duality, their optimal values are exactly equal, and every lower bound on polynomial degree can, in principle, be stated as a dual polynomial. The technique of dual polynomials has been used by Bun, Kothari, and Thaler [16] to prove strong lower bounds for a number of problems like k-distinctness, image size testing, and surjectivity. The first of them was later improved in [26]. Similarly strong adversary lower bounds for these problems are not known.

Since the adversary method is tight, for every polynomial lower bound, there ought to exist a similarly good adversary lower bound. However, a direct reduction was not known. In this paper, we prove a simple direct reduction, giving a mechanical way of converting every dual polynomial into an adversary lower bound of a specific form. We hope that this connection will give a better understanding of both techniques, and should enable their combined use, which could result in better lower bounds. Contrary to a large number of papers dealing with the general adversary method, all proofs in this paper are fairly elementary.

A related result is a direct reduction from the polynomial method to *multiplicative* adversary by Magnin and Roland [25], while we give a reduction to a more widely-used *additive* adversary. We also note that our construction has similarities to a recent powerful lower bound technique by Zhandry [34, 24]. It would be interesting to understand the connection between the two better.

The following result is the cornerstone of our reduction. Here, we consider the task of distinguishing whether an input $x \in [q]^n$ belongs to a set $X \subseteq [q]^n$ or $Y \subseteq [q]^n$. Informally, the result states that X and Y cannot be distinguished by a quantum query algorithm if they are *perfectly* indistinguishable by assignments of a corresponding weight. Recall that an *assignment* is a function $\alpha\colon S \to [q]$ defined on a subset S of the set of indices $[n]$. We write $x \sim \alpha$ if $x \in [q]^n$ *agrees* with the assignment α, that is, $x_j = \alpha(j)$ for all $j \in S$. The weight $|\alpha|$ of the assignment is the size of its domain S. It is possible to have an empty assignment $\emptyset$ of zero weight, in which case, every input string agrees to it.

▶ **Theorem 1.** *Let $X, Y \subseteq [q]^n$ be sets of inputs, and μ and ν be probability distributions on X and Y, respectively. Assume that, for any assignment α of weight $\le 2m$, we have the following perfect indistinguishability*

$$\Pr_{x\leftarrow\mu}[x \sim \alpha] = \Pr_{y\leftarrow\nu}[y \sim \alpha]. \tag{1}$$

That is, the probability that x agrees to α does not depend on whether x is sampled from μ or from ν. Then, the quantum query complexity of distinguishing X and Y is $\Omega(m)$.

The result itself is actually known. We will give two proofs in this paper. The first one in Section 2 uses the method of dual polynomials and it is purely for illustrative purposes. The second proof is new and is done using the adversary method. This is the main technical contribution of this paper. Let us give a very short outline here. The proof uses the following collection of vectors:

$$v_\alpha^X = \sum_{x\in X:x\sim\alpha} \sqrt{\mu_x}|x\rangle \qquad \text{and} \qquad v_\alpha^Y = \sum_{y\in Y:y\sim\alpha} \sqrt{\nu_y}|y\rangle,$$

where α is an assignment. By the indistinguishability, for every $k \le m$, there exists a linear isometry $W_{\le k}$ that maps v_α^X into v_α^Y for all α with $|\alpha| \le k$. The adversary matrix is $\Gamma = \sum_{k=0}^{m-1} W_{\le k}$. It is not hard to show it has norm m, and we prove that $\|\Gamma \circ \Delta_j\| \le 1$ for all j. This proof is contained in Sections 3 and 4. In Section 3, we only consider the space $\mathbb{R}^X$, and in Section 4, we substitute $\mathbb{R}^X$ with $\mathbb{R}^Y$ using indistinguishability.

In Section 5, we show how to use this result to transform a dual polynomial into an adversary bound. The idea is that a dual polynomial gives probability distributions μ and ν on two sets $\widetilde{X}$ and $\widetilde{Y}$ that are "close" to X and Y and that satisfy the promise of Theorem 1. We first prove the lower bound in the form of the adversary for distinguishing two probability distributions from [11], as we think it is conceptually closer to the dual polynomial. Obtaining a standard worst-case adversary bound is also easy.

2 Preliminaries

For a positive integer m, let $[m]$ denote the set $\{1, 2, ..., m\}$. For a predicate P, we write 1_P to denote the indicator variable that is 1 if P is true, and 0 otherwise.

We consider partial functions $f\colon D \to \{0,1\}$ with $D \subseteq [q]^n$. We denote $X = f^{-1}(1)$ and $Y = f^{-1}(0)$. Thus, the function f distinguishes X and Y. An element $x = (x_1, x_2, \dots, x_n) \in [q]^n$, is called an *input*, the set $[q]$ is called the *input alphabet*, and $x_j \in [q]$ are individual input symbols.

A *measure* on a finite set X is a function μ from X to the set of non-negative real numbers. We denote the value of μ on $x \in X$ by μ_x. The measure is a *probability distribution* if $\sum_{x\in X} \mu_x = 1$. We use $x \leftarrow \mu$ to denote that x is sampled from the probability distribution μ.[1]

[1] A more standard notation is $x \sim \mu$. But since we use $x \sim \alpha$ for agreement with an assignment α, we opted to use a different piece of notation.

2.1 Linear Algebra

An $X \times Y$ matrix is a matrix with rows labelled by the elements of X and columns by the elements of Y. The element of an $X \times Y$ matrix A at the intersection of the x-th row and the y-th column is denoted by $A[\![x, y]\!]$. For $X' \subseteq X$ and $Y' \subseteq Y$, the matrix $A[\![X', Y']\!]$ is the restriction of A to the rows in X' and the columns in Y'. We use $\|A\|$ to denote spectral norm of matrices. We identify a subspace and the corresponding orthogonal projector, which we usually denote by Π with additional decorations. An *isometry* is a linear operator that preserves inner product. We need the following well-known result:

▶ **Lemma 2.** *Assume $\mathcal{H}$ and $\mathcal{K}$ are two inner-product spaces. Let $(v_i)_{i\in A} \subseteq \mathcal{H}$ and $(w_i)_{i\in A} \subseteq \mathcal{K}$ be two collections of vectors indexed by the same index set A. Assume $\langle v_i, v_j\rangle = \langle w_i, w_j\rangle$ for all $i, j \in A$. Then, there exists an isometry $T\colon \operatorname{span}_i v_i \to \operatorname{span}_i w_i$ such that $Tv_i = w_i$ for all i.*

2.2 Adversary Bound

We use two different flavours of the negative-weighted adversary bound. Here we give the canonical version from [22] and later we state the distributional version from [11].

Assume we want to distinguish two sets of inputs $X, Y \subseteq [q]^n$ as above. Let Γ be a real $X \times Y$ matrix. For $j \in [n]$, denote by $\Gamma \circ \Delta_j$ the matrix of the same dimensions as Γ whose (x, y)-th entry is given by $\Gamma[\![x, y]\!] \cdot 1_{x_j \neq y_j}$. In other words, the entries with $x_j = y_j$ are being erased (replaced by zeroes).

▶ **Theorem 3** ([22]). *Assume that Γ is an $X \times Y$ real matrix such that $\|\Gamma \circ \Delta_j\| \le 1$ for all $j \in [n]$. Then, the (bounded-error) quantum query complexity of evaluating f is $\Omega(\|\Gamma\|)$.*

The matrix Γ from Theorem 3 is called the *adversary matrix*, and it is known that the bound of this theorem is tight [30].

As it can be guessed from the notation, the mapping $\Gamma \mapsto \Gamma \circ \Delta_j$ is usually expressed as an Hadamard product with a 01-matrix Δ_j of dimensions $X \times Y$. However, we find it more convenient to think of it as a mapping. In particular, we don't have to formally re-define the matrix Δ_j for matrices Γ of different dimensions, and the matrix Δ_j almost never appears by itself.

The norm of the matrix $\Gamma \circ \Delta_j$ is not always easy to estimate. The following trick from [12] based on [23, Fact 2.4] is of help here. With some stretch of notation, we write $\Gamma \overset{\Delta_j}{\longmapsto} B$ if $(\Gamma - B) \circ \Delta_j = 0$. In other words, we are allowed to arbitrary change the (x, y)-entries of Γ with $x_j = y_j$ in order to obtain B. Note that this is a relation, since B is not uniquely defined by Γ. The idea is as follows:

▶ **Proposition 4.** *For any B with $\Gamma \overset{\Delta_j}{\longmapsto} B$, we have $\|\Gamma \circ \Delta_j\| \le 2\|B\|$. Moreover, if f is a Boolean function, i.e., $D \subseteq \{0, 1\}^n$, then $\|\Gamma \circ \Delta_j\| \le \|B\|$.*

Hence, we can bound $\|\Gamma \circ \Delta_j\|$ from above by estimating $\|B\|$, which is often easier. We repeatedly use the following easy properties of thus defined relation Δ_j:

▶ **Proposition 5.** *For any $X \times Y$ matrices A, B, C, D and real numbers a and c, we have*

(a) *$A \overset{\Delta_j}{\longmapsto} A$ and $A \overset{\Delta_j}{\longmapsto} A \circ \Delta_j$;*

(b) *if $A \overset{\Delta_j}{\longmapsto} B$ and $C \overset{\Delta_j}{\longmapsto} D$, then $aA + cC \overset{\Delta_j}{\longmapsto} aB + cD$.*

2.3 Distributional Adversary

We also use the version of the adversary bound for distinguishing two probability distributions. This version is rather versatile as it allows the probability distributions to overlap and to have arbitrary acceptance probabilities.

▶ **Theorem 6** ([11]). *Assume $\mathcal{A}$ is a quantum algorithm that makes T queries to the input string $x = (x_1, \dots, x_n) \in [q]^n$, and performs a measurement at the end with two outcomes: "accept" or "reject". Let μ and ν be two probability distributions on $[q]^n$, and denote by s_μ and s_ν the acceptance probability of $\mathcal{A}$ when x is sampled from μ and ν, respectively. Then,*

$$T = \Omega\left(\min_{j\in[n]} \frac{\delta_\mu^* \Gamma \delta_\nu - \tau(s_\mu, s_\nu)\|\Gamma\|}{\|\Gamma \circ \Delta_j\|}\right), \tag{2}$$

for any $[q]^n \times [q]^n$ matrix Γ with real entries. Here,

$$\delta_\mu[\![x]\!] = \sqrt{\mu_x} \qquad \text{and} \qquad \delta_\nu[\![y]\!] = \sqrt{\nu_y} \tag{3}$$

are unit vectors in $\mathbb{R}^{[q]^n}$, and

$$\tau(s_\mu, s_\nu) = \sqrt{s_\mu s_\nu} + \sqrt{(1-s_\mu)(1-s_\nu)} \le 1 - \frac{|s_\mu - s_\nu|^2}{8}. \tag{4}$$

2.4 Polynomials

In the polynomial method, we have to assume that the function $f\colon D \to \{0,1\}$ is Boolean: $D \subseteq \{0,1\}^n$. If this does not hold, one has to make the function Boolean. A popular option is to introduce new variables $\widetilde{x_{i,a}}$ with $i \in [n]$ and $a \in [q]$, defined by $\widetilde{x_{i,a}} = 1_{x_i = a}$.

For $S \subseteq [n]$, the corresponding *character* is the function $\chi_S\colon \{0,1\}^n \to \{\pm 1\}$ defined by $\chi_S(x) = \prod_{i\in S}(-1)^{x_i}$. The characters form a basis of the space of functions $\mathbb{R}^{\{0,1\}^n}$. Hence, every function $f\colon \{0,1\}^n \to \mathbb{R}$ has a unique representation as a (multilinear) *polynomial*: $f = \sum_{S\subseteq[n]} \alpha_S \chi_S$. The size of the largest S with non-zero α_S is called the *degree* of f.

A degree-d *polynomial* is any function $p\colon \{0,1\}^n \to \mathbb{R}$ of degree at most d. A degree-d *dual polynomial* is a function $\phi\colon \{0,1\}^n \to \mathbb{R}$ satisfying

$$\sum_{x\in\{0,1\}^n} |\phi(x)| = 1 \qquad \text{and} \qquad \sum_{x\in\{0,1\}^n} \phi(x)\chi_S(x) = 0 \quad \text{for all } |S| \le d. \tag{5}$$

It is easy to check that the second condition above is equivalent to the following one:

$$\sum_{x\sim\alpha} \phi(x) = 0 \quad \text{for all assignments } \alpha \text{ with } |\alpha| \le d. \tag{6}$$

Dual polynomials [32] can be used to show inapproximability for real-valued total functions. We may assume $d < n$, since every function can be represented by a degree-d polynomial.

The proof of the following theorem, as well as that of Theorem 10 are based on linear programming duality, and are given in Appendix A for completeness.

▶ **Theorem 7.** *Let $d < n$. For any function $f\colon \{0,1\}^n \to \mathbb{R}$, we have*

$$\min_p \max_{x\in\{0,1\}^n} |f(x) - p(x)| = \max_\phi \sum_{x\in\{0,1\}^n} \phi(x) f(x), \tag{7}$$

where p ranges over all degree-d polynomials and ϕ ranges over all degree-d dual polynomials.

Let us now turn to the case of partial functions $f\colon D \to \{0,1\}$ with $D \subseteq \{0,1\}^n$. Again, we let $X = f^{-1}(1)$ and $Y = f^{-1}(0)$.

▶ **Definition 8.** We say that a polynomial $p\colon \{0,1\}^n \to \mathbb{R}$ ε-approximates a partial function $f\colon D \to \{0,1\}$ with $D \subseteq \{0,1\}^n$ if

- for every $x \in D$, we have $|p(x) - f(x)| \le \varepsilon$;
- for every $x \in \{0,1\}^n$, we have $0 \le p(x) \le 1$.

The importance of this definition stems from the following result:

▶ **Theorem 9** ([9]). *If a partial function $f\colon D \to \{0,1\}$ with $D \subseteq \{0,1\}^n$ can be evaluated by a T-query quantum algorithm with error at most ε, then f can be ε-approximated by a polynomial of degree at most $2T$.*

The corresponding analogue of Theorem 7 is slightly more involved. A similar result previously appeared in [18].

▶ **Theorem 10.** *The best approximation distance ε as in Definition 8 of the function f by a degree-d polynomial is given by*

$$\max\left\{ \max_{\phi} \left(\sum_{x\in X} \phi^+(x) - \sum_{x \notin Y} \phi^-(x) \right), 0 \right\}, \tag{8}$$

where the maximisation is over functions $\phi\colon \{0,1\}^n \to \mathbb{R}$ satisfying

$$\sum_{x\in X} \phi^+(x) + \sum_{x\in Y} \phi^-(x) = 1 \qquad \text{and} \qquad \sum_{x\in\{0,1\}^n} \phi(x)\chi_S(x) = 0 \quad \text{for all } |S| \le d. \tag{9}$$

Here $\phi^+(x) = \max\{0, \phi(x)\}$ and $\phi^-(x) = \max\{0, -\phi(x)\}$ are the positive and the negative parts of ϕ, respectively. We will still call ϕ a degree-d dual polynomial in this case, although it need not satisfy the first (normalisation) condition of (5).

Proof of Theorem 1 using Dual Polynomials. We may assume the function f is Boolean. It suffices to show that it cannot be approximated by a polynomial of degree less than $2m$. Let

$$\phi(x) = \begin{cases} \mu_x/2, & \text{if } x \in X; \\ -\nu_x/2, & \text{if } x \in Y; \\ 0, & \text{otherwise.} \end{cases}$$

This function satisfies (9) with $d = 2m$. Indeed, the first condition follows from μ and ν being probability distributions, and the second one follows from (6) since

$$\sum_{x\sim\alpha} \phi(x) = \frac{1}{2} \Pr_{x\leftarrow\mu}[x \sim \alpha] - \frac{1}{2} \Pr_{y\leftarrow\nu}[y \sim \alpha] = 0$$

by (1). The value of (8) is $1/2$, hence, by Theorem 10, it is impossible to get a better than trivial approximation. ◀

3 Δ-decomposition of $\mathbb{R}^X$

Let $X \subseteq [q]^n$ be a set of inputs, and let μ be some measure on X. We assume in this section that X is the support of μ, i.e., $\mu_x > 0$ for all $x \in X$. The goal of this section is to develop a decomposition of the space $\mathbb{R}^X$ convenient for the Δ_j operation and that takes into account the measure μ. Let us remind that we use the same notation, like $\Pi_{\le k}$, to denote *both* the subspace and the corresponding orthogonal projector.

3.1 Definition of subspaces

For each assignment α, define the following vector in $\mathbb{R}^X$:

$$v_\alpha = \sum_{x \sim \alpha} \sqrt{\mu_x} |x\rangle. \tag{10}$$

Based on these vectors, we define a number of subspaces. First, for $k \in \{0, 1, \dots, n\}$:

$$\Pi_{\le k} = \operatorname*{span}_{\alpha\colon |\alpha| = k} v_\alpha. \tag{11}$$

▷ Claim 11. We have $\Pi_{\le k-1} \subseteq \Pi_{\le k}$ and $\Pi_{\le n} = \mathbb{R}^X$.

Proof. Let α be an assignment of weight $k-1$, and i be an element of $[n]$ outside the domain of α. Then,

$$v_\alpha = \sum_{a \in [q]} v_{\alpha \cup \{i \mapsto a\}},$$

proving the first claim.

For the second claim, note that an assignment α of weight n defines an individual input. ◁

This gives an orthogonal decomposition of $\mathbb{R}^X$ into subspaces

$$\Pi_k = \Pi_{\le k} \cap \Pi_{\le k-1}^\perp = \Pi_{\le k} - \Pi_{\le k-1}.$$

3.2 Example

A simple example is $X = [q]^n$ with the uniform distribution μ_x. Define two orthogonal projectors on $\mathbb{R}^q$: $E_0 = J_q/q$ and $E_1 = I_q - E_0$, where J_q is the all-1 matrix. Then,

$$\Pi_k = \sum_{s \in \{0,1\}^n : |s| = k} E_{s_1} \otimes E_{s_2} \otimes \dots \otimes E_{s_n},$$

where $|s|$ is the Hamming weight. These operators are similar to the ones used in the construction of the adversary lower bound for element distinctness [10] and sum-problems [12].

Note that while the vectors v_α in (10) only have non-negative entries, the projectors $\Pi_{\le k}$ can have negative entries. For instance, such are matrices $\Pi_{\le 1}$ in the above example.

3.3 Action of Δ_j

Let us consider the action of Δ_j with $j \in [n]$. For that, we define the following variant of the above subspaces $\Pi_{\le k}$:

$$\Xi_{\le k, \circ j} = \operatorname*{span}_{\alpha\colon |\alpha| = k,\, \alpha \text{ defined on } j} v_\alpha.$$

In particular, we again have $\Xi_{\le n, \circ j} = \mathbb{R}^X$. This time, however, $\Xi_{\le 0, \circ j}$ is the empty subspace.

▷ Claim 12. We have the following:

(a) $\Xi_{\le k-1, \circ j} \subseteq \Xi_{\le k, \circ j}$;
(b) $\Pi_{\le k-1} \subseteq \Xi_{\le k, \circ j} \subseteq \Pi_{\le k}$;
(c) $\Delta_j \circ \Xi_{\le k, \circ j} = 0$.

Proof. The proof of (a) is analogous to the proof of Claim 11.

The second inclusion of (b) holds because $\Xi_{\le k,\circ j}$ is a span of a subset of vectors of $\Pi_{\le k}$. To prove the first inclusion of (b), it suffices to show that an arbitrary v_α with $|\alpha| = k-1$ is contained in $\Xi_{\le k,\circ j}$. The proof of that is analogous to the proof of Claim 11. However, this time we take $i = j$ if α is not defined on j (and an arbitrary i as before, otherwise).

Now let us prove (c). Note that $\Xi_{\le k,\circ j}$ can be written as a direct sum

$$\Xi_{\le k,\circ j} = \bigoplus_{b\in[q]} \Xi_{\le k, j\mapsto b} \tag{12}$$

of orthogonal projectors

$$\Xi_{\le k,j\mapsto b} = \operatorname*{span}_{\alpha\colon |\alpha|=k,\ \alpha(j)=b} v_\alpha.$$

Each $\Xi_{\le k,j\mapsto b}$ acts on the subspace spanned by $x \in X$ with $x_j = b$. Hence, $\Delta_j \circ \Xi_{\le k,j\mapsto b} = 0$. By linearity, $\Delta_j \circ \Xi_{\le k,\circ j} = 0$. ◁

3.4 Standard Form of Adversary

As a warm-up for the next sections, we describe the following “standard” form of the “adversary” matrix on $\mathbb{R}^X$:

$$\sum_{k=0}^{m-1} \Pi_{\le k} = \sum_{k=0}^{m} (m-k)\Pi_k. \tag{13}$$

Clearly, the norm of this matrix is m. The action of Δ_j is defined as

$$\sum_{k=0}^{m-1} \Pi_{\le k} \overset{\Delta_j}{\longmapsto} \sum_{k=0}^{m-1} \bigl(\Pi_{\le k} - \Xi_{\le k,\circ j}\bigr), \tag{14}$$

where we use Proposition 5 and point (c) of Claim 12.

▷ Claim 13. The norm of the operator on the right-hand side of (14) is 1.

Proof. The operator in question is the sum of projectors $\Pi_{\le k} - \Xi_{\le k,\circ j}$. By point (b) of Claim 12, we know that $\Pi_{\le k}$ is contained in $\Xi_{\le k+1,\circ j}$. Hence, these projectors are pairwise orthogonal, and the norm of the operator is 1. ◁

In the following section, we will transfer this construction for $X \times Y$-matrices.

4 Second Proof of Theorem 1

Here, we give a proof of Theorem 1, which is based on the adversary method. In this section, we use upper indices X and Y in the following way. If the upper index X is used, the corresponding object is equal to the one without the upper index as defined in Section 3. If the upper index Y is used, we use the same object but with the probability distribution ν on Y instead of μ on X. For example:

$$v_\alpha^X = \sum_{x\in X: x\sim\alpha} \sqrt{\mu_x}|x\rangle \in \mathbb{R}^X \qquad \text{and} \qquad v_\alpha^Y = \sum_{y\in Y: y\sim\alpha} \sqrt{\nu_y}|y\rangle \in \mathbb{R}^Y.$$

Similarly, $\Pi^X_{\le k}$ and $\Xi^X_{\le k,\circ j}$ are projectors in $\mathbb{R}^X$, and $\Pi^Y_{\le k}$ and $\Xi^Y_{\le k,\circ j}$ are projectors in $\mathbb{R}^Y$.

▶ **Lemma 14.** *In the assumptions of Theorem 1, there exists a linear isometry* $W\colon \Pi^Y_{\le m} \to \Pi^X_{\le m}$ *that maps* v^Y_α *into* v^X_α *for each* $|\alpha| \le m$.

Proof. Let α and β be assignments of weight at most m. Note that

$$\langle v^X_\alpha, v^X_\beta \rangle = \Pr_{x\sim\mu}[x \sim \alpha \wedge x \sim \beta] = \Pr_{y\sim\nu}[y \sim \alpha \wedge y \sim \beta] = \langle v^Y_\alpha, v^Y_\beta \rangle.$$

Indeed, either α and β contradict each other, in which case the both sides of the above equality are zero, or they can be merged into one assignment of weight at most $2m$, in which case (1) applies. Hence, by Lemma 2, there exists a linear isometry W that maps v^Y_α into v^X_α for each $|\alpha| \le m$. ◀

The following theorem defines the adversary matrix Γ which, when plugged into Theorem 3, gives Theorem 1. This matrix will be important in the next section for the reduction from the polynomial method.

▶ **Theorem 15.** *In the assumptions of Theorem 1, the following* $X \times Y$ *matrix*

$$\Gamma = W\left(\sum_{k=0}^{m-1} \Pi^Y_{\le k}\right). \tag{15}$$

has the following properties:

(a) *its norm is* m*, as witnessed by* $\|\Gamma\| = \delta^*_\mu \Gamma \delta_\nu = m$ *with* δ_μ *and* δ_ν *as defined in (3); and*

(b) *the action of* Δ_j *is given by*

$$\Gamma \overset{\Delta_j}{\longmapsto} W\left(\sum_{k=0}^{m-1} \left(\Pi^Y_{\le k} - \Xi^Y_{\le k,\circ j}\right)\right), \tag{16}$$

where the norm of the matrix on the right-hand side is 1.

Proof. Eq. (13) gives the decomposition of $\sum_{k=0}^{m-1} \Pi^Y_{\le k}$ into eigenspaces since the subspaces Π^Y_k are pairwise orthogonal. The maximal eigenvalue m is achieved on Π^Y_0, which is spanned by $v^Y_\emptyset = \delta_\nu$, where $\emptyset$ denotes the empty assignment. Also, W is an isometry that maps $v^Y_\emptyset$ into $v^X_\emptyset = \delta_\mu$, which proves point (a) of the theorem.

The validity of the action of Δ_j in (16) follows from the claim that $\Delta_j \circ (W\Xi^Y_{\le k,\circ j}) = 0$. The proof of this claim is similar to the point (c) of Claim 12. We use decomposition (12) for $\Xi^Y_{\le k,\circ j}$, and observe that the range of $W\Xi^Y_{\le k,j\mapsto b}$ is $\Xi^X_{\le k,j\mapsto b}$. Hence, $\Delta_j \circ (W\Xi^Y_{\le k,j\mapsto b}) = 0$, and the first half of point (b) follows by linearity.

The second half of point (b) follows from Claim 13 and the fact that W is an isometry. ◀

5 Reduction from Dual Polynomial to Adversary

In this section, we demonstrate a direct conversion of a polynomial lower bound into an adversary lower bound. We do so by taking a dual polynomial that witnesses degree at least d and convert it into an adversary bound of value $\Omega(d)$.

For warm-up, we consider the case of total functions in Section 5.1, and then the general case of partial functions in Section 5.2. In both cases, we use the distributional version of the adversary bound, Theorem 6, which we find conceptually more appropriate in this case. However, it is not hard to reduce to the usual version of the bound, Theorem 3, as well, which we do in Section 5.3.

5.1 Total Functions

We start with the case when $f\colon \{0,1\}^n \to \{0,1\}$ is a total Boolean function. Assume it cannot be 1/3-approximated by a polynomial of degree d. In this case, we can use Theorem 7. Let ϕ be a degree-d dual polynomial attaining the maximum in (7). Thus,

$$\sum_{x\in\{0,1\}^n} \phi(x)f(x) \ge 1/3. \tag{17}$$

Our goal is to prove an adversary lower bound of $\Omega(d)$.

Let us define

$$\widetilde{X} = \{x \in \{0,1\}^n \mid \phi(x) \ge 0\} \qquad \text{and} \qquad \widetilde{Y} = \{y \in \{0,1\}^n \mid \phi(y) < 0\}, \tag{18}$$

and two measures

$$\mu\colon \widetilde{X} \to \mathbb{R},\, x \mapsto 2\phi(x) \qquad \text{and} \qquad \nu\colon \widetilde{Y} \to \mathbb{R},\, y \mapsto -2\phi(y).$$

From (6) applied to empty α, we get that $\sum_x \phi(x) = 0$. Also, $\sum_x |\phi(x)| = 1$. Hence,

$$\sum_{x\in\widetilde{X}} \mu_x = \sum_{y\in\widetilde{Y}} \nu_y = 1, \tag{19}$$

that is, both μ and ν are probability distributions.

Using (6) again, we get that for each assignment α of weight at most d, we have

$$\Pr_{x\leftarrow\mu}[x \sim \alpha] = \Pr_{y\leftarrow\nu}[y \sim \alpha]. \tag{20}$$

Thus, by Theorem 1, the quantum query complexity of distinguishing $\widetilde{X}$ and $\widetilde{Y}$ is $\Omega(d)$. This is a nice development, but we would really like to prove the same result for the sets $X = f^{-1}(1)$ and $Y = f^{-1}(0)$. Luckily, by condition (17), these sets are sufficiently well correlated.

Define the $\widetilde{X} \times \widetilde{Y}$ matrix $\widetilde{\Gamma}$ as in (15) with the sets $\widetilde{X}$ and $\widetilde{Y}$, the distributions μ and ν, and $m = d/2$. By Theorem 15, we have

$$\|\widetilde{\Gamma}\| = \delta_\mu^* \widetilde{\Gamma} \delta_\nu = d/2, \qquad \text{and} \qquad \|\widetilde{\Gamma} \circ \Delta_j\| \le 1 \quad \text{for all } j \in [n]. \tag{21}$$

Since Theorem 6 requires an $\{0,1\}^n \times \{0,1\}^n$ adversary matrix, we extend $\widetilde{\Gamma}$ with zeroes to fit this requirement.

▶ **Proposition 16.** *In the above notations, Theorem 6 with the adversary matrix $\widetilde{\Gamma}$ and the distributions μ and ν gives an $\Omega(d)$ lower bound on the number of queries made by any quantum query algorithm $\mathcal{A}$ that distinguishes X and Y with error probability at most 1/6.*

Proof. Note that (17) is equivalent to

$$\sum_{x\in f^{-1}(1)} \mu_x - \sum_{y\in f^{-1}(1)} \nu_y \ge 2/3.$$

This is the difference between the "ideal" acceptance probabilities of $\mathcal{A}$ on μ and ν, i.e, in the hypothetical case when the algorithm never errs. Since the actual error of the algorithm $\mathcal{A}$ is at most 1/6, we get

$$s_\mu - s_\nu \ge 1/3$$

in notations of Theorem 6. From (4), we get that $\tau(s_\mu, s_\nu) \le 1 - \Omega(1)$. Pluging this and (21) into (2), we get that the query complexity of $\mathcal{A}$ is $\Omega(d)$. ◀

5.2 Partial Functions

Now let us consider the case of partial functions $f\colon D \to \{0,1\}$ with $D \subseteq \{0,1\}^n$. Again, we assume that the best optimisation distance by a degree-d polynomial is more than 1/3. Let ϕ be the optimal degree-d dual polynomial from Theorem 10. Then we have from (8):

$$\sum_{x\in X} \phi^+(x) - \sum_{x\notin Y} \phi^-(x) \geq 1/3. \tag{22}$$

The sets $\widetilde{X}$ and $\widetilde{Y}$ are still defined as in (18). By (9), we still have that

$$\sum_{x\in\{0,1\}^n} \phi^+(x) = \sum_{x\in\{0,1\}^n} \phi^-(x).$$

But, in order to define μ and ν, we have to choose a different scaling factor. We have

$$\sum_{x\in\{0,1\}^n} \phi^-(x) = \sum_{x\in Y} \phi^-(x) + \sum_{x\notin Y} \phi^-(x) \leq \sum_{x\in Y} \phi^-(x) + \sum_{x\in X} \phi^+(x) - 1/3 = 2/3,$$

where we used (22) and the first condition from (9). Let us denote the left-hand side of the above inequality by M. Then, we can define probability distributions

$$\mu\colon \widetilde{X} \to \mathbb{R},\, x \mapsto \phi(x)/M \qquad \text{and} \qquad \nu\colon \widetilde{Y} \to \mathbb{R},\, y \mapsto -\phi(y)/M. \tag{23}$$

So that (22) becomes

$$\sum_{x\in X} \mu_x - \sum_{y\notin Y} \nu_y \geq 1/2. \tag{24}$$

The equation (20) still holds, and we use the same construction of $\widetilde{\Gamma}$, which still satisfies (21).

Let $\mathcal{A}$ be an algorithm that evaluates f with error ε. Denote by p_x the acceptance probability of the algorithm on an input $x \in \{0,1\}^n$. So, we have $p_x \geq 1-\varepsilon$ for $x \in X$, $p_x \leq \varepsilon$ for $x \in Y$, and $0 \leq p_x \leq 1$ for all x. Thus,

$$\begin{aligned} s_\mu - s_\nu &= \sum_{x\in X} \mu_x p_x + \sum_{x\notin X} \mu_x p_x - \sum_{y\in Y} \nu_y p_y - \sum_{y\notin Y} \nu_y p_y \\ &\geq (1-\varepsilon)\sum_{x\in X} \mu_x - \varepsilon\sum_{y\in Y} \nu_y - \sum_{y\notin Y} \nu_y \geq \sum_{x\in X} \mu_x - \sum_{y\notin Y} \nu_y - 2\varepsilon \geq 1/2 - 2\varepsilon \geq 1/4, \end{aligned}$$

assuming $\varepsilon \leq 1/8$.

In the same way as in Section 5.1, Theorem 6 implies that the query complexity of $\mathcal{A}$ is $\Omega(d)$.

5.3 Usual Version of the Adversary

In this section, we obtain a usual version of the adversary bound from a dual polynomial. Let us recap the construction.

We assume $f\colon D \to \{0,1\}$ with $D \subseteq \{0,1\}^n$ is a partial Boolean function, where we define $X = f^{-1}(1)$ and $Y = f^{-1}(0)$. Assume ϕ is a degree-d dual polynomial that satisfies (9) of Theorem 10 and attains value at least 1/3 in (8).

Let $\widetilde{X}, \widetilde{Y} \subseteq \{0,1\}^n$ be as in (18), and μ and ν be probability distributions in (23). They satisfy (20), therefore, we can apply Theorem 15 with $m = d/2$, and obtain an $\widetilde{X} \times \widetilde{Y}$-matrix $\widetilde{\Gamma}$ as in (15) for the sets $\widetilde{X}$ and $\widetilde{Y}$ with the probability distributions μ and ν on them. This matrix satisfies (21). We extend it with zeroes to form an $\{0,1\}^n \times \{0,1\}^n$-matrix, which we still denote $\widetilde{\Gamma}$.

▶ **Theorem 17.** *In the above assumptions, the $X \times Y$-matrix*

$$\Gamma = \widetilde{\Gamma}[\![X, Y]\!]$$

satisfies $\|\Gamma\| = \Omega(d)$ *and* $\|\Gamma \circ \Delta_j\| \le 1$ *for all* $j \in [n]$.

Proof. As Γ is a sub-matrix of $\widetilde{\Gamma}$, we get $\|\Gamma \circ \Delta_j\| \le 1$ for all j from (21). By the same equation (21), it suffices to show that $\|\Gamma\| = \Omega\big(\|\widetilde{\Gamma}\|\big)$.

We know that $\widetilde{\Gamma}\delta_\nu = \|\widetilde{\Gamma}\|\delta_\mu$ by point (a) of Theorem 15. This gives us

$$\Big\|\widetilde{\Gamma}[\![X, \{0,1\}^n]\!]\ \delta_\nu\Big\| = \|\widetilde{\Gamma}\| \cdot \big\|\delta_\mu[\![X]\!]\big\|.$$

On the other hand,

$$\Big\|\widetilde{\Gamma}[\![X, \{0,1\}^n]\!]\ \delta_\nu\Big\| \le \Big\|\widetilde{\Gamma}[\![X, Y]\!]\ \delta_\nu[\![Y]\!]\Big\| + \Big\|\widetilde{\Gamma}[\![X, \overline{Y}]\!]\ \delta_\nu[\![\overline{Y}]\!]\Big\| \le \big\|\widetilde{\Gamma}[\![X, Y]\!]\big\| + \|\widetilde{\Gamma}\| \cdot \big\|\delta_\nu[\![\overline{Y}]\!]\big\|,$$

where $\overline{Y} = \{0,1\}^n \setminus Y$. Thus,

$$\big\|\widetilde{\Gamma}[\![X, Y]\!]\big\| \ge \|\widetilde{\Gamma}\|\Big(\big\|\delta_\mu[\![X]\!]\big\| - \big\|\delta_\nu[\![\overline{Y}]\!]\big\|\Big) = \|\widetilde{\Gamma}\| \frac{\big\|\delta_\mu[\![X]\!]\big\|^2 - \big\|\delta_\nu[\![\overline{Y}]\!]\big\|^2}{\big\|\delta_\mu[\![X]\!]\big\| + \big\|\delta_\nu[\![\overline{Y}]\!]\big\|}.$$

From (24), we get that

$$\big\|\delta_\mu[\![X]\!]\big\|^2 - \big\|\delta_\nu[\![\overline{Y}]\!]\big\|^2 = \sum_{x \in X} \mu_x - \sum_{y \notin Y} \nu_y \ge 1/2.$$

Also, $\big\|\delta_\mu[\![X]\!]\big\| + \big\|\delta_\nu[\![\overline{Y}]\!]\big\| \le 2$, hence, we obtain

$$\big\|\widetilde{\Gamma}[\![X, Y]\!]\big\| \ge \frac{1}{4}\|\widetilde{\Gamma}\|,$$

as required. ◀

References

1 Scott Aaronson, Shalev Ben-David, and Robin Kothari. Separations in query complexity using cheat sheets. In *Proc. of 48th ACM STOC*, pages 863–876, 2016. doi:10.1145/2897518.2897644.

2 Scott Aaronson, Shalev Ben-David, Robin Kothari, Shravas Rao, and Avishay Tal. Degree vs. approximate degree and quantum implications of Huang's sensitivity theorem. In *Proc. of 53rd ACM STOC*, pages 1330–1342, 2021. doi:10.1145/3406325.3451047.

3 Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *Journal of the ACM*, 51(4):595–605, 2004. doi:10.1145/1008731.1008735.

4 Andris Ambainis. Quantum lower bounds by quantum arguments. *Journal of Computer and System Sciences*, 64(4):750–767, 2002. doi:10.1006/jcss.2002.1826.

5 Andris Ambainis. Polynomial degree vs. quantum query complexity. In *Proc. of 44th IEEE FOCS*, pages 230–239, 2003. doi:10.1109/SFCS.2003.1238197.

6 Andris Ambainis and Aleksandrs Belovs. An exponential separation between quantum query complexity and the polynomial degree. In *Proc. of 38th IEEE CCC*, volume 264 of *LIPIcs*, pages 24:1–24:13. Dagstuhl, 2023. doi:10.4230/LIPIcs.CCC.2023.24.

7 Howard Barnum and Michael Saks. A lower bound on the quantum query complexity of read-once functions. *Journal of Computer and System Sciences*, 69(2):244–258, 2004. doi:10.1016/j.jcss.2004.02.002.

8 Howard Barnum, Michael Saks, and Mario Szegedy. Quantum decision trees and semi-definite programming. In *Proc. of 18th IEEE CCC*, pages 179–193, 2003. doi:10.1109/CCC.2003.1214419.

9 Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, 2001. doi:10.1145/502090.502097.

10 Aleksandrs Belovs. Adversary lower bound for element distinctness, 2012. arXiv:1204.5074.

11 Aleksandrs Belovs, Gilles Brassard, Peter Høyer, Marc Kaplan, Sophie Laplante, and Louis Salvail. Provably secure key establishment against quantum adversaries. In *Proc. of 12th TQC*, volume 73 of *LIPIcs*, pages 3:1–3:17. Dagstuhl, 2018. doi:10.4230/LIPIcs.TQC.2017.3.

12 Aleksandrs Belovs and Ansis Rosmanis. On the power of non-adaptive learning graphs. *Computational Complexity*, 23(2):323–354, 2014. doi:10.1007/s00037-014-0084-1.

13 Aleksandrs Belovs and Robert Špalek. Adversary lower bound for the k-sum problem. In *Proc. of 4th ACM ITCS*, pages 323–328, 2013. doi:10.1145/2422436.2422474.

14 Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997. doi:10.1137/S0097539796300933.

15 Harry Buhrman and Robert Špalek. Quantum verification of matrix products. In *Proc. of 17th ACM-SIAM SODA*, pages 880–889, 2006. arXiv:quant-ph/0409035.

16 Mark Bun, Robin Kothari, and Justin Thaler. The polynomial method strikes back: Tight quantum query bounds via dual polynomials. In *Proc. of 50th ACM STOC*, pages 297–310, 2018. doi:10.1145/3188745.3188784.

17 Mark Bun and Justin Thaler. Dual lower bounds for approximate degree and Markov-Bernstein inequalities. *Information and Computation*, 243:2–25, 2015. doi:10.1016/j.ic.2014.12.003.

18 Mark Bun and Justin Thaler. Dual polynomials for collision and element distinctness. *Theory of Computing*, 12(16):1–34, 2016. doi:10.4086/toc.2016.v012a016.

19 Sebastian Dörn and Thomas Thierauf. The quantum query complexity of algebraic properties. In *Proc. of 16th FCT*, volume 4639 of *LNCS*, pages 250–260. Springer, 2007. doi:10.1007/978-3-540-74240-1_22.

20 Christoph Dürr, Mark Heiligman, Peter Høyer, and Mehdi Mhalla. Quantum query complexity of some graph problems. In *Proc. of 31st ICALP*, volume 3142 of *LNCS*, pages 481–493. Springer, 2004. doi:10.1007/978-3-540-27836-8_42.

21 Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proc. of 28th ACM STOC*, pages 212–219, 1996. doi:10.1145/237814.237866.

22 Peter Høyer, Troy Lee, and Robert Špalek. Negative weights make adversaries stronger. In *Proc. of 39th ACM STOC*, pages 526–535, 2007. doi:10.1145/1250790.1250867.

23 Troy Lee, Rajat Mittal, Ben W. Reichardt, Robert Špalek, and Mario Szegedy. Quantum query complexity of state conversion. In *Proc. of 52nd IEEE FOCS*, pages 344–353, 2011. doi:10.1109/FOCS.2011.75.

24 Qipeng Liu and Mark Zhandry. On finding quantum multi-collisions. In *Proc. of 38th EUROCRYPT*, volume 11478 of *LNCS*, pages 189–218, 2019. doi:10.1007/978-3-030-17659-4_7.

25 Loïck Magnin and Jérémie Roland. Explicit relation between all lower bound techniques for quantum query complexity. In *Proc. of 30th STACS*, volume 20 of *LIPIcs*, pages 434–445. Dagstuhl, 2013. doi:10.4230/LIPIcs.STACS.2013.434.

26 Nikhil S. Mande, Justin Thaler, and Shuchen Zhu. Improved approximate degree bounds for k-distinctness. In *Proc. of 15th TQC*, volume 158 of *LIPIcs*, pages 2:1–2:22, 2020. doi:10.4230/LIPIcs.TQC.2020.2.

27 Noam Nisan. CREW PRAMs and decision trees. *SIAM Journal on Computing*, 20(6):999–1007, 1991. doi:10.1137/0220062.

28 Noam Nisan and Mario Szegedy. On the degree of Boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994. doi:10.1007/BF01263419.

29 Ben W. Reichardt. Span programs and quantum query complexity: The general adversary bound is nearly tight for every Boolean function. In *Proc. of 50th IEEE FOCS*, pages 544–551, 2009. doi:10.1109/FOCS.2009.55.

30 Ben W. Reichardt. Reflections for quantum query algorithms. In *Proc. of 22nd ACM-SIAM SODA*, pages 560–569, 2011. doi:10.1137/1.9781611973082.44.

31 Ben W. Reichardt and Robert Špalek. Span-program-based quantum algorithm for evaluating formulas. *Theory of Computing*, 8:291–319, 2012. doi:10.4086/toc.2012.v008a013.

32 Alexander A Sherstov. The pattern matrix method. *SIAM Journal on Computing*, 40(6):1969, 2011. doi:10.1137/080733644.

33 Alexander A. Sherstov. Approximating the and-or tree. *Theory of Computing*, 9(20):653–663, 2013. doi:10.4086/toc.2013.v009a020.

34 Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In *Proc. of 39th CRYPTO*, volume 11693 of *LNCS*, pages 239–268, 2019. doi:10.1007/978-3-030-26951-7_9.

35 Shengyu Zhang. On the power of Ambainis lower bounds. *Theoretical Computer Science*, 339(2):241–256, 2005. doi:10.1016/j.tcs.2005.01.019.

A Linear Programming for Dual Polynomials

A.1 Proof of Theorem 7

The left-hand side of (7) is equal to the optimal value of the following linear optimisation problem:

$$\begin{aligned}
\text{minimise} \quad & \varepsilon \\
\text{subject to} \quad & f(x) - \sum_S \alpha_S \chi_S(x) \le \varepsilon && \text{for all } x \in \{0,1\}^n; && \text{(25a)} \\
& f(x) - \sum_S \alpha_S \chi_S(x) \ge -\varepsilon && \text{for all } x \in \{0,1\}^n; && \text{(25b)} \\
& \alpha_S \in \mathbb{R} && \text{for all } S \subseteq [n],\ |S| \le d; \\
& \varepsilon \in \mathbb{R}.
\end{aligned}$$

Let us write the Lagrangian with the dual variables $a_x \ge 0$ for (25a) and $b_x \ge 0$ for (25b):

$$\varepsilon - \sum_x a_x \Bigl(\varepsilon - f(x) + \sum_S \alpha_S \chi_S(x)\Bigr) - \sum_x b_x \Bigl(\varepsilon + f(x) - \sum_S \alpha_S \chi_S(x)\Bigr) \tag{26}$$

Let us denote $\phi(x) = a_x - b_x$, so that we can rewrite the last expression as

$$\sum_x \phi(x) f(x) + \varepsilon \Bigl(1 - \sum_x a_x - \sum_x b_x\Bigr) - \sum_S \alpha_S \Bigl(\sum_x \phi(x)\chi_S(x)\Bigr). \tag{27}$$

In the dual optimisation problem, all of the brackets in (27) must be zero.

We can turn any dual polynomial into a feasible solution to the dual (27) by taking $a_x = \phi^+(x)$ and $b_x = \phi^-(x)$.

For the opposite direction, consider optimal primal and dual solutions, whose values are equal due to strong duality. If $\varepsilon > 0$, then, by complementary slackness, at most one of a_x and b_x is non-zero for each x, therefore, $|\phi(x)| = a_x + b_x$. Hence, ϕ is a dual polynomial satisfying $\sum_x \phi(x) f(x) = \varepsilon$. If $\varepsilon = 0$, we can take ϕ equal to the normalised parity function.

A.2 Proof of Theorem 10

In this case, we have the following linear programming problem:

$$
\begin{aligned}
&\text{minimise} && \varepsilon \\
&\text{subject to} && \sum_S \alpha_S\chi_S(x) \ge 1-\varepsilon && \text{for all } x\in X; && \text{(28a)}\\
& && \sum_S \alpha_S\chi_S(x) \le \varepsilon && \text{for all } x\in Y; && \text{(28b)}\\
& && \sum_S \alpha_S\chi_S(x) \ge 0 && \text{for all } x\notin X; && \text{(28c)}\\
& && \sum_S \alpha_S\chi_S(x) \le 1 && \text{for all } x\notin Y; && \text{(28d)}\\
& && \alpha_S\in\mathbb{R} && \text{for all } S\subseteq[n],\ |S|\le d;\\
& && \varepsilon\in\mathbb{R}.
\end{aligned}
$$

Let us write the Lagrangian with the dual variables $a_x, b_x, c_x, d_x \ge 0$ for (28a) – (28d), respectively:

$$
\begin{aligned}
\varepsilon \quad - \quad \sum_{x\in X} a_x\Bigl(\sum_S \alpha_S\chi_S(x) - 1 + \varepsilon\Bigr) \quad - \quad \sum_{x\in Y} b_x\Bigl(\varepsilon - \sum_S \alpha_S\chi_S(x)\Bigr)\\
- \quad \sum_{x\notin X} c_x\Bigl(\sum_S \alpha_S\chi_S(x)\Bigr) \quad - \quad \sum_{x\notin Y} d_x\Bigl(1 - \sum_S \alpha_S\chi_S(x)\Bigr)
\end{aligned}
\tag{29}
$$

Let us define

$$
\phi(x) = \begin{cases} a_x - d_x & \text{if } x\in X;\\ c_x - b_x & \text{if } x\in Y;\\ c_x - d_x & \text{if } x\notin X\cup Y.\end{cases}
$$

Then, we can rewrite (29) as

$$
\sum_{x\in X} a_x - \sum_{x\notin Y} d_x + \varepsilon\Bigl(1 - \sum_{x\in X} a_x - \sum_{x\in Y} b_x\Bigr) - \sum_S \alpha_S\Bigl(\sum_{x\in\{0,1\}^n} \phi(x)\chi_S(x)\Bigr). \tag{30}
$$

Again, in the dual optimisation problem, all the brackets in (30) must be zero.

If ϕ satisfies (9), then we can take $a_x = \phi^+(x)$ for $x\in X$, $b_x = \phi^-(x)$ for $x\in Y$, $c_x = \phi^+(x)$ for $x\notin X$, and $d_x = \phi^-(x)$ for $x\notin Y$, and get a feasible solution to the dual.

For the opposite direction, consider optimal primal and dual solutions, whose values are equal by strong duality. We may assume $\varepsilon > 0$. By complementary slackness, for each x, at most one of the dual variables is non-zero, except for the case when $\varepsilon = 1/2$, in which case both a_x and b_x can be non-zero. Either way, we get $a_x = \phi^+(x)$ for $x\in X$, $b_x = \phi^-(x)$ for $x\in Y$, and $d_x = \phi^-(x)$ for $x\notin Y$. Thus we obtain the required dual formulation of Theorem 10.

Let us note that maximisation with 0 is required in (8). For example, consider the case $d = n-1$, and X and Y are of size 1. The function can be approximated by a polynomial of degree at most 1, thus $\varepsilon = 0$. On the other hand, by the second condition of (9), ϕ must be equal to a multiple of the parity function. It is easy to see that $\sum_{x\in X}\phi^+(x) - \sum_{x\notin Y}\phi^-(x)$ is actually negative in this case.

Quantum Delegation with an Off-The-Shelf Device

Anne Broadbent
Department of Mathematics and Statistics, University of Ottawa, Canada
Nexus for Quantum Technologies, University of Ottawa, Canada

Arthur Mehta
Department of Mathematics and Statistics, University of Ottawa, Canada
Nexus for Quantum Technologies, University of Ottawa, Canada

Yuming Zhao
Institute for Quantum Computing, University of Waterloo, Canada
Department of Pure Mathematics, University of Waterloo, Canada

Abstract

Given that reliable cloud quantum computers are becoming closer to reality, the concept of delegation of quantum computations and its verifiability is of central interest. Many models have been proposed, each with specific strengths and weaknesses. Here, we put forth a new model where the client trusts only its classical processing, makes no computational assumptions, and interacts with a quantum server in a *single* round. In addition, during a set-up phase, the client specifies the size n of the computation and receives an untrusted, *off-the-shelf (OTS)* quantum device that is used to report the outcome of a single measurement.

We show how to delegate polynomial-time quantum computations in the OTS model. This also yields an interactive proof system for all of QMA, which, furthermore, we show can be accomplished in statistical zero-knowledge. This provides the first relativistic (one-round), two-prover zero-knowledge proof system for QMA.

As a proof approach, we provide a new self-test for n EPR pairs using only constant-sized Pauli measurements, and show how it provides a new avenue for the use of simulatable codes for local Hamiltonian verification. Along the way, we also provide an enhanced version of a well-known stability result due to Gowers and Hatami and show how it completes a common argument used in self-testing.

2012 ACM Subject Classification Theory of computation → Interactive proof systems; Theory of computation → Quantum complexity theory

Keywords and phrases Delegated quantum computation, zero-knowledge proofs, device-independence

Digital Object Identifier 10.4230/LIPIcs.TQC.2024.12

Related Version *Full Version*: https://arxiv.org/abs/2304.03448 [9]

Funding This work was supported by the Mitacs Accelerate program IT24833 in collaboration with industry partner Agnostiq, online at https://agnostiq.ai.
Anne Broadbent: A.B. is supported by the Air Force Office of Scientific Research under award number FA9550-20-1-0375, NSERC, and the University of Ottawa's Research Chairs program.

Acknowledgements We thank Seyed Sajjad Nezhadi, Gregory Rosenthal, William Slofstra, Jalex Stark, and Henry Yuen for discussions on the model. We thank Alex Grilo for discussions on some of our proof techniques. We also thank Thomas Vidick for detailed discussions on the formulation and proof of Theorem 19. We thank the anonymous reviewers for suggesting the use case of an OTS for a classical proof system.

19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024).
Editors: Frédéric Magniez and Alex Bredariol Grilo; Article No. 12; pp. 12:1–12:23
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In an interactive proof system, a computationally-bounded verifier interacts with a powerful prover in order to verify the truthfulness of an agreed-upon problem instance. Starting with QMA, and followed by QIP and QMIP (among others), *quantum* interactive proof system, (in which the verifier is *quantum* polynomial-time) were defined and studied [48, 49, 30].

Yet, these quantizations depend crucially on the tacit assumption that the verifier has access to *trusted* quantum polynomial-time verification. Given the current state-of-the-art in quantum computation development, the inherent difficulty at characterizing quantum systems, and the fact that there is no way to reliably verify the trace of a quantum computation, there is ample evidence that this assumption may be questionable. Indeed, despite impressive technological improvements, we may ultimately have to contend with a reality where quantum computers are never as trustworthy or reliable as classical devices. This prospect has motivated consideration of models where the verifier has access to very limited but trusted quantum functionality [1, 4, 18], or where the verifier is entirely classical and the prover is computationally bounded [31], while another class called MIP^* models an efficient classical verifier interacting with several isolated, unbounded quantum provers [14]. Each approach provides advantages and encounters challenges: early quantum servers will be expensive and thus all else equal, requiring a single prover is preferable; on the other hand, existing single-prover protocols either require a trusted device or make computational assumptions. Multi-prover protocols utilize powerful device-independence techniques which avoid these assumptions but at the high cost of requiring several powerful provers and requiring isolation.

The current zeitgeist in this field allows for imaginative considerations of how we describe and model tasks in a quantum world. These approaches have in common that instead of considering the straightforward quantum analog of classical protocols, we strive to make considerations that are naturally motivated in the quantum setting[1]. Here, we continue on this momentum and introduce a novel approach to proof verification, where the set-up itself can only be motivated in the quantum setting. To this end, we consider the following question:

▶ **Question 1.** What is the expressive power of the class of **relativistic**, interactive proof systems with a single quantum prover, and a classical verifier having access to an **off-the-shelf untrusted quantum device**?

Off-the-shelf Device. We call the above model the *off-the-shelf (OTS)* model since it models the fact that the verifier, in addition to interacting with a standard prover, has access to a device that is (1) generic (it does not depend on the instance of the problem to be solved, only on the instance size), (2) efficient (for completeness, polynomial resources suffice), (3) completely untrusted (for soundness, there are no assumptions on its computational power or inner-workings). Importantly, *relativistic* refers to a 1-round protocol; this is desirable for its relative ease in enforcing isolation[2].

Operationally, we imagine the OTS model as the prover providing the verifier with such a generic, off-the-shelf device ahead of the proof verification. In particular, the preparation of such a device in terms of its capabilities is independent of the particular problem instance, although we do allow dependence on its size. Once in possession of this device, the verifier

[1] See, for instance, the recent work on the complexity of preparing quantum states and unitaries [42].

[2] A relativistic protocol is highly desirable in the multi-prover scenario since isolation can be enforced using relative position and response times [10, 22].

may query the prover and simultaneously use a single measurement from the off-the-shelf device, which leads the verifier to *accept* or *reject*. The figures of merit for the interactive proof system are the usual *completeness* and *soundness*.

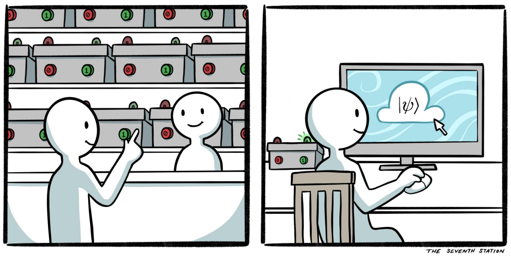


Figure 1 During the set-up the verifier selects an off-the-shelf device based on the required size of the problem instance. Afterward, the verifier is free to select any language and instance and interacts in a single round with both the prover and the off-the-shelf device, leading to the *accept/reject* output of the verifier.

Since the OTS scenario models aspects of near-term proof verification using untrusted quantum devices, we naturally wish to understand how it relates to some of the most relevant and studied properties of **interactive proof systems**:

▶ **Question 2.** Can the OTS model provide novel approaches to **zero-knowledge proof systems** and to **delegated quantum computation**?

Classical and Quantum Interactive Proof Systems. In the model of interactive proof systems (IP), an efficient classical verifier interacts with an all-powerful and untrusted prover in order to verify the correctness of a statement [20]. We note that class NP corresponds to a single-message interaction (with MA being in probabilistic version), while AM incorporates a single round (*i.e.*, *two* messages).

In *a multiprover interactive proof system* (MIP), a verifier interacts with multiple isolated provers [3]. Each of the models above has been *quantized*, *i.e.*, extended to the setting where some (or all) of the parties are quantum. This is captured, *e.g.* by the classes QMA (the quantum version of MA), QIP (the quantum version of IP) and MIP* (a version of a multi-prover interactive proof system (MIP) where the unbounded provers share entanglement). Groundbreaking results have characterized some these quantum classes, *e.g.* QIP $=$ PSPACE [24] and MIP* $=$ RE [27].

Zero-Knowledge Proof Systems. A strong motivation for the study of interactive proof systems is the connections to the counter-intuitive concept of a zero-knowledge proof system [19, 2]. Informally, a proof system is *zero-knowledge* when the verifier is unable to learn anything beyond the fact that the agreed-upon instance is true. This is more formally treated by establishing the existence of a *simulator* which can reproduce the transcript of the interaction.

Zero-knowledge proof systems were first extended to the quantum setting by Watrous [50], who considered the setting where the verifier has access to a trusted polynomial-time quantum device. Subsequently, it was shown that under certain cryptographic assumptions, all problems in QMA admit a zero-knowledge proof system [7, 8, 5] (again, assuming the verifier has trusted polynomial-time quantum computation). There have been several approaches in the case of a fully classical verifier. Vidick and Zhang showed that argument protocols can be made to satisfy the zero-knowledge property [47]. Recent work by Crépeau and Stuart [17] provides a two-prover one-round zero-knowledge proof system for NP. The work of Chiesa, Forbes, Gur and Spooner provides a two-prover zero-knowledge proof system for NEXP [12], however, their work requires polynomially many rounds of interaction. Work due to Grilo, Yuen, and Slofstra [23] shows that any proof system for MIP* can be made zero-knowledge at the cost of adding four additional provers. Although these works provide inspiration for studying zero-knowledge proof systems in the OTS model, as far as we are aware, they do not directly contribute to our main question on ZK. In fact, according to the current state-of-the-art, an implicit open question [22] is the following: *"Does there exists a* relativistic *zero-knowledge proof system for* QMA *with two provers and a classical verifier?"*. We emphasize that our OTS model takes this question further, by requiring one of the provers to operate generically and independently of the problem instance.

Delegated Quantum Computation. Delegated quantum computation allows a computationally-weak classical client to delegate a computational task to an untrusted, polynomial-time quantum server. Under certain conditions, an interactive proof system leads in a straightforward way to a protocol for delegated quantum computation. Typically, this is achieved if the interactive proof system captures *e.g.* QMA, and furthermore, given the witness, the prover is efficient; it is also relevant that the QMA witness is used in such a way that we can scale down the proof system in order to achieve a delegation protocol for BQP (*e.g.* [22])[3]. The sketch above is also applicable to the scenario of multiple servers. Note that because of the resemblance between the models of the interactive proof system and delegated quantum computation, we occasionally confound the two – using the complexity class acronym to refer to the interaction pattern between prover(s) and verifier – but we emphasize that in delegated quantum computation *protocols*, the server is always computationally bounded (as opposed to a prover in interactive *proof systems*).

Following Reichardt, Unger, and Vazirani [41], who showed a delegated quantum computation for the setting of MIP*, much progress was made, aiming at improving parameters and techniques; despite these efforts, as far as we are aware, none of the existing works are applicable to our model. Notable here is the work of [16] which uses *quasilinear* resources for *both* servers, and achieves at best a constant round complexity, as well as [22] which is the first 2-server, 1-round (relativistic) protocol for delegated quantum computations, but uses the full polynomial-power of both servers.

1.1 Summary of results

In this work, we make important steps towards answering the above questions:

- We show that any language in QMA has a statistical ZK proof system in the OTS model.
- We show that the above OTS proof system can be adapted for delegated quantum computation for any problem in BQP, while remaining ZK and in the OTS model.

We now give more details and motivation for our model and an overview of our main contributions at the conceptual level.

Model. As introduced earlier, we are interested in modeling near-term proof verification and delegation of quantum computations. To this end, we propose a new paradigm that is particularly relevant to the quantum scenario: a verifier having access to an OTS device. To

[3] BQP is closed under complementation, hence this is sufficient for delegation

motivate the model, consider that the complexity class QMA models a verifier having access to fully-trusted polynomial-time quantum computation. While such a verifier is *skeptical* of the prover (and thus needs to verify the claimed proof independently), in the quantum case, a new level of skepticism is possible, namely that the verifier's quantum processing is untrusted. A common solution in this case is to postulate *two* (or more) untrusted and all-powerful devices together with a classical verifier; this is the realm of MIP^*. In this work, we propose a new paradigm that treats the provers asymmetrically. Starting with a conventional two-prover interactive proof system, we ask that only *one* of the provers do the heavy lifting (via its unbounded computational capabilities), with the second prover becoming efficient and completely generic (for completeness, this prover need not even be given a description of the task at hand; soundness, however, is shown against *two* unbounded provers).

The inspiration for our model finds its roots in the elegant 2-prover, 1-round protocol introduced by Grilo [22]. The approach Grilo uses is a game that involves an energy test and the Pauli braiding test to verify the shared EPR pairs' integrity and the accuracy of Pauli-X/Z measurements on local terms H_i. Although in [22] the analysis assumes that players Alice and Bob are randomly assigned roles, it is natural to consider an asymmetric version of this game where one prover's functionality can be made independent of the problem instance. Grilo's work motivates the formal introduction and study of a new class which one may expect to be lower-bounded by QMA. As we outline in Section 1.2 there are substantial technical obstructions to extending this game to obtain a zero-knowledge protocol.

We denote OTS the set of all languages L that can be decided under a constant completeness-soundness gap, in the model that follows. Before the instance $x \in L$ is selected, the classical verifier is provided with an untrusted off-the-shelf device which only depends on a parameter n, indicating the size of the problem instance (without loss of generality, we can assume that the prover provides such OTS device). For completeness, such a device shares an entangled state $|\psi\rangle_n$ with a quantum prover and will be purported to perform efficient measurements from a predetermined list of available options[4]. The verifier may select any choice $x \in L$ provided $|x| \leq n$ and *simultaneously* uses a single question to the prover and to the device; the verifier then determines whether or not to accept based on the responses. We stress that OTS proof systems are sound against both an unbounded prover and unbounded OTS.

We observe that OTS is a refinement of and thus contained in MIP^*, and is also a generalization of AM, where the otherwise classical verifier has additional 1-round query access to a small, off-the-shelf quantum device. In summary, we have $\mathsf{AM} \subseteq \mathsf{OTS} \subseteq \mathsf{MIP}^*$ (see also Figure 2).

In a classical proof system, an OTS can be understood as an instance-independent hardware token. This device can be used to provide a commitment for a zero-knowledge proof system for NP [19]; what is more, the one-time property of the OTS can be used as an oblivious transfer device, which then yields a non-interactive zero-knowledge proof system for NP [28]. We note that in the quantum case, our model requires a fully classical verifier and hence the case of zero-knowledge for QMA [7, 5] in the OTS model is much more complex, and a classical-verifier analogue to the NP proof systems above is not directly applicable. Other approaches based on using the OTS as a one-time memory [6] also run into a roadblock due to the fact that we require a fully classical verifier.

[4] The entangled state $|\psi_n\rangle$ is consumed during the interactive proof, hence a new OTS must be obtained for subsequent evaluations (equivalently, the entanglement must eventually be replenished). This situation is entirely analogous to the case of shared randomness which is also consumed in an interactive proof system and must also eventually be replenished.

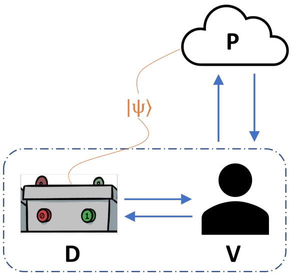


Figure 2 Off-the-shelf (OTS) proof system. P is the quantum prover, V is the classical verifier, and D a rudimentary off-the-shelf device which is entangled with the prover; each arrow represents a single classical message.

OTS Proof Systems for QMA. Our first result is that any language in QMA is also in OTS.

▶ **Theorem 3** (Restated as part of Theorem 31)**.** QMA $\subseteq$ OTS.

An interpretation of this result is that starting with a conventional proof system for QMA, we can exchange the unwavering trust of the verifier in its quantum verification process for a classical verifier with two new features: (1) the verifier has access to an untrusted, and instance-independent, off-the-shelf quantum device; and (2) the verifier interacts with the prover (and the device) in a single simultaneous round.

Zero-knowledge OTS Proof System for QMA. What is more, we show that the OTS proof system for QMA is also *statistical zero-knowledge*, meaning that we can simulate in classical polynomial time the verifier's transcript when interacting with the provers on a yes-instance.

▶ **Theorem 4** (Restated in Theorem 31)**.** *For every language L in* QMA*, there exists a statistical zero-knowledge OTS proof system for L.*

Delegated Quantum Computation in the OTS Model. As our final conceptual contribution, we show how our OTS proof system for QMA (Theorem 3) can be adapted to the setting of delegated quantum computation; note that the ZK property as described above also extends to the delegated quantum computation paradigm.

▶ **Theorem 5** (Restated as Theorem 32)**.** BQP *has a relativistic delegated quantum computation protocol in the OTS model with the statistical zero-knowledge property.*

We believe that this result is of particular impact since it addresses a new model for delegated quantum computation that has distinct conceptual benefits over existing protocols:

1. Comparing to single-server protocols, we note that we make an extra assumption of an off-the-shelf, isolated device. However, the benefits are:
   a. We achieve soundness against an unbounded server; existing single-server, classical-client delegation protocols require computational assumptions [31].
   b. The client does not trust *any* quantum device at all; existing single-server, statistically secure protocols require trust in a small quantum preparation device [4, 18].

2. Comparing to existing multiple-server (MIP^*) protocols, we note that:
   a. Our approach only requires a single high-performance quantum server that handles the bulk of the computations; with a secondary efficient and generic device which need not even be given a description of the problem instance. This has practical advantages, especially when we consider that the off-the-shelf device can be acquired ahead of the verification stage (Figure 1).
   b. Our approach is a single round, which means that relativistic means to enforce isolation are possible. The only other known relativistic protocol requires full quantum computational power for both servers and is not ZK [22].

1.2 Proof approach and technical contributions

We first introduce two important techniques.

Self-testing. *Self-testing* (also called *device-independence*) is a ubiquitous and powerful technique in the study of MIP^* and related delegation protocols. The concept was introduced by Mayers and Yao [33]. Informally, a protocol *self-tests* a particular state or measurement when this state/measurements (or an equivalent version thereof) are required for obtaining the maximal acceptance probability. The most well-known examples are the non-local games known as the CHSH game and the Magic Square game [13, 36, 40, 45]. Subsequently, numerous works have enriched our understanding of self-testing and its applications to delegated quantum computation, *e.g.*, [34, 35, 15, 11, 16, 37, 38]. Current approaches to formalizing self-testing use the theory of approximate representation theory of groups and C^*-algebras [43, 44, 32]. These formalisms, and especially their operationally-useful *approximate* versions utilize a key stability result due to Gowers and Hatami which allows one to relate approximate representations to exact representations [21].

Simulatable Codes. Recent works by Grilo, Yuen, and Slofstra [23], as well as Broadbent and Grilo [5] introduce the notion of simulatable codes as a tool for establishing zero-knowledge proof systems and protocols in the quantum setting. The idea is to use techniques from quantum error-correcting codes to create a "simulatable" witness or proof for use in the verification process. Here the witness is *simulatable* in the sense that there is an efficient classical algorithm which can reproduce the description of the local density matrix of the witness on any small enough subspace. This is a pivotal tool in establishing zero-knowledge, and the application of the technique consists in developing a verification protocol, (or verification circuit in the case of [5]) which verifies such simulatable witnesses; this can then be applied to the situation of encoding *e.g.*, a witness for QMA into a simulatable code [5].

1.2.1 Obstructions to the straightforward approach

In delegating quantum computations in two- or multi-server models, the classical verifier is able to command quantum provers [41] using two intertwined tests: (1) a computational test, with acceptance probability based on the required quantum computation (*e.g.*, computation-by-teleportation [41] or energy checking of a local Hamiltonian [26, 22]); (2) a rigidity test, ensuring provers' actions stay within a known range (*e.g.*, self-test via CHSH game or Pauli braiding test). In order to establish the ZK property, we must show that responses from the provers can be simulated using a classical probabilistic polynomial-time (PPT) device. Generally, approaches used for the rigidity test can be simulated in a straightforward way, hence the difficulty in obtaining ZK in this setting is in simulating the energy test. Furthermore, even if both tests are simulatable in isolation, this does not guarantee the ZK property since a malicious verifier may form question pairs emanating from different tests, during a single round.

Grilo [22] presents a game $\mathcal{G}(H)$ determined by an "XZ-type"[5] Hamiltonian H. Honest provers for this game share suitably many EPR pairs, and one prover privately holds a ground state for H. The game $\mathcal{G}(H)$ combines an energy test with the Pauli braiding test [37, 46]. During the energy test, one prover reports measurement results of a randomly chosen term H_i on their side of EPR pairs, and the other provides teleportation keys from a Bell basis measurement on the other EPR pairs and the ground state. Combining the energy test with the Pauli braiding test allows the verifier to ensure that provers share n EPR pairs and that the required Pauli-X/Pauli-Z measurements are performed when measuring the local term H_i.

The straightforward approach to obtaining a two-prover ZK proof system would be to combine recent results on simulatable codes in order to make the measurement results in Grilo's energy test simulatable. More specifically, one could apply the well-known circuit-to-Hamiltonian construction using the family of simulatable verification circuits given in [5]. Given such a circuit V, it is shown that local measurements on the ground state of the corresponding Hamiltonian H_V are simulatable and thus this approach would make the results of the energy test simulatable. Unfortunately, this approach fails for two technical reasons.

The Choice of Encoding. Firstly, one cannot employ previously-known self-testing techniques to show the players perform the required measurements on the simulatable ground states given in [5]. On the one hand, previously-studied single-round self-testing techniques can only be used to show the players perform Pauli-X, and Pauli-Z measurements. On the other hand, the choice of physical gates used by Broadbent and Grilo during the encoding of logic gates may result in a local Hamiltonian that is not of XZ-type and thus local terms H_i may require measurements that have no known self-test.

The Size of the Measurement. The second obstruction arises from the fact that existing rigidity tests in this setting require both players to make large-sized measurements on their shared state. These large measurements can provide an avenue for attack by a malicious verifier which compromises the zero-knowledge property. In particular, since the Pauli braiding test allows for requests for measurements on all qubits, a malicious verifier may indicate to one player that an energy test is being played and simultaneously request Pauli-X and Pauli-Z measurements on a large number of qubits. Such a measurement result cannot be simulated using simulatable codes, which only protect against constant-sized measurements, and thus this compromises zero-knowledge.

1.2.2 Overview of proof and technical results

In order to correct for an appropriate choice of encoding, we prove that one can re-instantiate the verification circuit given by Broadbent and Grilo using an approach to simulatable codes given in [23]. This change allows us to encode logical gates of the verification circuit given

[5] These are Hamiltonians where each local term H_i is a real linear combination of tensor products of the Pauli-X and Pauli-Z operators.

by Broadbent and Grilo using a different set of physical gates and consequently, we show that the local Hamiltonian corresponding to the circuit is of XZ-type, while preserving simulatability.

▶ **Theorem 6** (Informal version of Theorem 27). *For any language $L = (L_{yes}, L_{no})$ in* QMA*, there is a family of verification circuits V_x satisfying (1) the circuit-to-Hamiltonian construction applied to V_x produces a Hamiltonian H_x which is of XZ-type, and (2) if $x \in L_{yes}$ there exists a polynomial-time algorithm that can approximate the reduced density matrix obtained by tracing out all but 6 qubits of the ground state of H_x.*

In order to overcome the large measurement problem, we introduce a new self-test called the *low-weight Pauli braiding test* (LWPBT) which can self-test the low-weight tensor products of Pauli measurements and n EPR pairs but only requires the players to make measurements on a constant number of qubits.

▶ **Theorem 7** (Informal version of Theorem 18). *The low-weight Pauli braiding test can self-test for n EPR pairs and 6-qubit Pauli measurements. This self-test is robust in the sense that any ε-perfect strategy must be poly$(n)\sqrt{\varepsilon}$ close to the canonical strategy.*

We use a group-theoretical approach to prove the rigidity of the LWPBT. It can be shown that the canonical perfect strategy $\widetilde{S}$ for LWPBT defines an irreducible representation of the Weyl-Heisenberg group H and every near-perfect strategy S for LWPBT forms an approximate homomorphism f of H. The well-known Gowers-Hatami theorem [21] and its variant [46] imply that the approximate homomorphism f of a finite group is close to a representation ϕ, so S must be close to $\widetilde{S}$. However, some subtle mathematical problems have come up in earlier approaches. In particular, one may need to discard some irreducible constituents of ϕ that do not correspond to $\widetilde{\mathcal{S}}$. To tackle this problem, we make further improvements to the state-of-art understanding of the stability of groups. In particular, in Theorem 18 we state and prove an enhanced version of the Gowers-Hatami theorem that can be used for the stability analysis of the Weyl-Heisenberg group. Aside from our use case, this new version can simplify previous approaches to self-testing.

We use the above technical results to derive a modified version of [22] by interleaving the following tests: (1) a computational test consisting of an energy test in which a simulatable witness uses low-weight Pauli-X and Pauli-Z measurements and, (2) a rigidity test consisting of the LWPBT. The result of this modified Grilo protocol gives a ZK OTS protocol with an inverse polynomial completeness-soundness gap. Finally, we apply a threshold parallel repetition theorem to the above protocol to amplify the completeness-soundness gap to be constant, thus demonstrating both Theorem 3 and Theorem 4. We then show that the proof system is of a form that can be scaled down to yield a delegation protocol, yielding Theorem 5.

2 Preliminaries

We take $[n]$ to denote the set $\{1, \dots n\}$. Given two real valued functions $f, g : \mathbb{R} \to \mathbb{R}$, we write $f = O(g)$ (resp. $f = \Omega(g)$) if there exists a positive real number M and an $x_0 \in \mathbb{R}$ such that $|f(x)| \leq Mg(x)$ (resp. $|f(x)| \geq Mg(x)$) for all $x \geq x_0$. We call a function f negligible, and write $f = \text{negl}(n)$, if for all constants $c > 0$ we have $f = O(n^{-c})$. For two distributions P and Q on a finite set $\mathcal{X}$ the statistical differences of P and Q is given by $\sum_{x \in \mathcal{X}} |P(x) - Q(x)|$.

In this paper, all Hilbert spaces are finite-dimensional. Given a Hilbert space $\mathcal{H}$, we use $\mathcal{B}(\mathcal{H})$ to denote the set of bounded linear operators acting on $\mathcal{H}$, use $\mathcal{U}(\mathcal{H})$ to denote the group of unitary operators on $\mathcal{H}$, and use $\mathbb{1}_{\mathcal{H}}$ to denote the identity operator on $\mathcal{H}$. Given an operator $A \in \mathcal{B}(\mathcal{H})$ we take A^* to denote the adjoint operator (equivalently the conjugate transpose) and define the trace norm $\|A\|_{tr} := \text{Tr}\sqrt{A^* A}$.

2.1 Quantum information

A quantum state ρ on $\mathcal{H}$ is a positive operator in $\mathcal{B}(\mathcal{H})$ with $\mathrm{Tr}(\rho) = 1$. It induces a semi-norm $\|A\|_\rho := \sqrt{\mathrm{Tr}(A^* A\rho)}$ on $\mathcal{B}(\mathcal{H})$ which we call the ρ-norm. This norm is left unitarily invariant, meaning that $\|UA\|_\rho = \|A\|_\rho$ for all $U \in \mathcal{U}(\mathcal{H})$ and $A \in \mathcal{B}(\mathcal{H})$. Given two quantum states ρ and σ we define their trace distance $D(\rho, \sigma) = \frac{1}{2}\|\rho - \sigma\|_{tr} = \max_P \mathrm{Tr}(P(\rho - \sigma))$ where the max is taken over all projections $P \in \mathcal{B}(\mathcal{H})$.

We use $|\Phi_{\text{EPR}}\rangle$ to denote the EPR pair in $\mathbb{C}^2 \otimes \mathbb{C}^2$ and use $|\Phi_{\text{EPR}}^{\otimes n}\rangle$ to denote the n-qubit EPR pair. We also take σ_I, σ_X, and σ_Z to denote the following Pauli operators:

$$\sigma_I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \sigma_X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \text{ and } \sigma_Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{1}$$

For every $a \in \{0,1\}^n$ and $W \in \{I, X, Z\}^n$, we use $\sigma_W(a)$ to denote the operator $\otimes_{i\in[n]}\sigma_{W_i}^{a_i}$ on $(\mathbb{C}^2)^{\otimes n}$ where $\sigma_I^0 = \sigma_X^0 = \sigma_Z^0 = \sigma_I$. Definitions of these gates and other fundamental concepts from quantum computing can be found in [39].

Families of Quantum Circuits. A *unitary quantum circuit* is simply a unitary which can be written as a product of gates from some universal gate set $\mathcal{U}$. Unless otherwise specified we will assume the universal gate set is the following universal gate set $\{H, \Lambda(X), \Lambda^2(X)\}$, where H is the Hadamard gate, $\Lambda(X)$ is the controlled σ_X gate, and $\Lambda^2(X)$ is the Toffoli gate. A *general quantum circuit* or simply a *quantum circuit* is a unitary quantum circuit that can additionally apply non-unitary gates which, introduce qubits initialized in the 0 state, trace out qubits, or measure qubits in the standard basis.

▶ **Definition 8** (Polynomial-time uniform circuit family). *We say a family of quantum circuits $\{Q_n\}_{n\in\mathbb{N}}$ is a polynomial-size family of quantum circuits if there exists polynomial r such that Q_n has size at most $r(n)$. A family of quantum circuits $\{Q_n\}$ is called* polynomial-time uniform *family if there exists a polynomial time Turing machine that on input 1^n outputs a description of Q_n. In this case, the family will also be a polynomial-size family of quantum circuits.*

Given a quantum circuit Q, we denote its size (number of gates and number of wires) by $|Q|$. The task of delegating the computation of Q is captured by the following promise problem:

▶ **Definition 9** (Q-CIRCUIT). *The input is a quantum circuit Q on n qubits. The problem is to decide between the following two cases:*

- ***Yes.*** $\|((|1\rangle\langle 1| \otimes I_{n-1})Q|0^n\rangle\|^2 \geq 1 - \gamma$
- ***No.*** $\|((|1\rangle\langle 1| \otimes I_{n-1})Q|0^n\rangle\|^2 \leq \gamma$

when we are promised that one of the two cases holds.

Problem in Definition 9 is known to be BQP-complete for $1 - 2\gamma > \frac{1}{\mathrm{poly}(n)}$.

2.2 Non-local games and rigidity

A two-player[6] one-round nonlocal game $\mathcal{G}$ is a tuple $\left(\lambda, \mu, \mathcal{I}_A, \mathcal{I}_B, \mathcal{O}_A, \mathcal{O}_B\right)$, where $\mathcal{I}_A, \mathcal{I}_B$ are finite input sets, and $\mathcal{O}_A, \mathcal{O}_B$ are finite output sets, μ is a probability distribution on $\mathcal{I}_A \times \mathcal{I}_B$, and $\lambda : \mathcal{O}_A \times \mathcal{O}_B \times \mathcal{I}_A \times \mathcal{I}_B \to \{0,1\}$ determines the win/lose conditions. A

[6] These two players are commonly called Alice and Bob.

quantum strategy $\mathcal{S}$ for $\mathcal{G}$ is given by finite-dimensional Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$, a unit vector $|\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B$, Alice's POVMs $\{E^x_a : a \in \mathcal{O}_A\}, x \in \mathcal{I}_A$ on $\mathcal{H}_A$, and Bob's POVMs $\{F^y_b : b \in \mathcal{O}_B\}, y \in \mathcal{I}_B$ on $\mathcal{H}_B$. The winning probability of $\mathcal{S}$ for game $\mathcal{G}$ is given by

$$\omega(\mathcal{G}, \mathcal{S}) := \sum_{a,b,x,y} \mu(x,y)\lambda(a,b|x,y) \langle\psi| E^x_a \otimes F^y_b |\psi\rangle .$$

A quantum strategy $\mathcal{S}$ for a non-local game $\mathcal{G}$ is said to be *perfect* if $\omega(\mathcal{G}, \mathcal{S}) = 1$. When the game is clear from the context we simply write $\omega(\mathcal{S})$ to refer to the winning probability. The *quantum value* of a non-local game $\mathcal{G}$ is defined as

$$\omega^*(\mathcal{G}) := \sup\{\omega(\mathcal{S}) : \mathcal{S} \text{ a quantum strategy for } G\}.$$

In this paper, we assume all measurements employed in a quantum strategy are PVMs. An m-outcome PVM $\{P_1, \cdots, P_m\}$ corresponds to an observable $\sum_{j\in[m]} \exp(\frac{2\pi i}{m} j)P_j$, so a quantum strategy for a game $\mathcal{G} = \big(\lambda, \mu, \mathcal{I}_A, \mathcal{I}_B, \mathcal{O}_A, \mathcal{O}_B\big)$ can also be specified by a triple

$$\mathcal{S} = (\tau^A, \tau^B, |\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B)$$

where $\tau^A(x)$, $x \in \mathcal{I}_A$ are $\mathcal{O}_A$-outcome observables on $\mathcal{H}_A$, and $\tau^B(y)$, $y \in \mathcal{I}_B$ are $\mathcal{O}_B$-outcome observables on $\mathcal{H}_B$.

Here we introduce the well-known Mermin-Peres Magic Square game, in which Alice and Bob are trying to convince the verifier that they have a solution to a system of equations over $\mathbb{Z}_2$. There are 9 variables $v_1, \ldots, v_9$ in a 3×3-array whose rows are labeled r_1, r_2, r_3 and columns are labeled c_1, c_2, c_3.

Each row or column corresponds to an equation: variables along the rows or columns in $\{r_1, r_2, r_3, c_1, c_2\}$ sum to 0; variables along the column c_3 sum to 1. In each round, Bob receives one of the 6 possible equations and he must respond with a satisfying assignment to the given equation. Alice is then asked to provide a consistent assignment to one of the variables contained in the equation Bob received. The following table describes an operator solution for this system of equations:

Table 1 Operator solution for Magic Square game.

$A_1 = \sigma_I \otimes \sigma_Z$	$A_2 = \sigma_Z \otimes \sigma_I$	$A_3 = \sigma_Z \otimes \sigma_Z$
$A_4 = \sigma_X \otimes \sigma_I$	$A_5 = \sigma_I \otimes \sigma_X$	$A_6 = \sigma_X \otimes \sigma_X$
$A_7 = \sigma_X \otimes \sigma_Z$	$A_8 = \sigma_Z \otimes \sigma_X$	$A_9 = \sigma_X\sigma_Z \otimes \sigma_Z\sigma_X$

2.3 Complexity classes and zero knowledge

▶ **Definition 10** (QMA). *A promise problem $L = (L_{yes}, L_{no})$ is in* QMA *if there exist polynomials p and q, and a polynomial-time uniform family of quantum circuits $\{Q_n\}$ where Q_n takes as input a string $x \in \Sigma^*$ with $|x| = n$, a $p(n)$-qubit quantum state $|\psi\rangle$, and $q(n)$ auxiliary qubits in state $|0\rangle^{\otimes q(n)}$, such that:*

- *(Completeness) if $x \in L_{yes}$, then there exists some $|\psi\rangle$ such that Q_n accepts $(x, |\psi\rangle)$ with probability at least $1 - \mathrm{negl}(n)$, and*
- *(Soundness) if $x \in L_{no}$, then for any state $|\psi\rangle$, Q_n accepts $(x, |\psi\rangle)$ with probability at most* $\mathrm{negl}(n)$.

We sometimes refer to the family of circuits $\{Q_n\}$ in Definition 10 simply as a family of *verification circuits.*

The following *local Hamiltonian problem* is QMA-complete for parameters $k = 5$ and $\beta - \alpha = \frac{1}{\mathrm{poly}(n)}$ [29].

▶ **Definition 11.** *Let $k \in \mathbb{N}$, $\alpha, \beta \in \mathbb{R}$ with $\alpha < \beta$, the k-*Local Hamiltonian *problem with parameters α and β is the following promise problem. Let n be the number of qubits of a quantum system. The input is a set of $m(n)$ Hamiltonians $H_1, \ldots, H_{m(n)}$ where m is a polynomial in n and each H_i acts on k qubits out of the n qubit system with $\|H_i\| \leq 1$. For $H = \sum_{j=1}^{m(n)} H_j$ the promise problem is to decide between the following.*

- ***Yes.*** *There exists an n-qubit state $|\varphi\rangle$ such that $\langle\varphi| H |\varphi\rangle \leq a \cdot m(n)$.*
- ***No.*** *For every n-qubit state $|\varphi\rangle$ it holds that $\langle\varphi| H |\varphi\rangle \geq b \cdot m(n)$.*

In this work, we also deal with MIP^* proof systems that involve two provers and one round.

▶ **Definition 12.** *A promise language $L = (L_{yes}, L_{no})$ is in $\mathsf{MIP}^*[2,1]_{c,s}$ if there exists a polynomial-time computable function that takes an instance $x \in L$ to a description of a non-local game $\mathcal{G}_x$ satisfying the following conditions.*

- *(Completeness) For every $x \in L_{yes}$ we have $\omega^*(\mathcal{G}_x) \geq c$.*
- *(Soundness) For every $x \in L_{no}$ we have $\omega^*(\mathcal{G}_x) < s$.*

We refer to the mapping, $x \mapsto \mathcal{G}_x$, as a $\mathsf{MIP}^*[2,1]_{c,s}$ proof system, or in some places a $\mathsf{MIP}^*[2,1]_{c,s}$ protocol. When the parameters are clear from the context we simply call it an MIP^* proof system.

Next, we discuss *zero knowledge*. In an interactive proof system, a malicious verifier $\widehat{V}$ is a probabilistic polynomial-time Turing machine which on input x and randomness θ samples question q_1 for either Alice or Bob. Given reply r_1, the malicious verifier samples question q_2 in a way that may depend on q_1 and r_1. For a given quantum strategy $\mathcal{S}$ and malicious verifier $\widehat{V}$, we take $View(\widehat{V}(x), \mathcal{S})$ to be the random variable corresponding to the transcript of questions and answers $(x, \theta, q_1, r_1, q_2, r_1)$. A protocol is zero-knowledge when for all "yes" instances a simulator can sample from the distribution above.

▶ **Definition 13.** *An* $\mathrm{MIP}^*[2,1]_{c,s}$ *proof system is* statistical zero-knowledge *if for every $x \in L_{yes}$ there exists an honest prover strategy $\mathcal{S}$ satisfying the following:*

1. *$\omega^*(\mathcal{S}) \geq c$.*
2. *For any PPT malicious verifier $\widehat{V}$ there exists a PPT simulator $Sim_{\widehat{V}}$ with output distribution that is ε-close to $View(\widehat{V}(x), S)$ in statistical distance for some negligible function $\varepsilon(|x|)$.*

2.4 Simulatable codes and encodings of gates

Recall that a quantum error-correcting code (QECC) $\mathcal{C} = [[n,k]]$ is a map $\mathrm{Enc} : (\mathbb{C}^2)^{\otimes k} \to (\mathbb{C}^2)^{\otimes n}$, which encodes a k-qubit state $|\psi\rangle$ into an n-qubit state $\mathrm{Enc}(|\psi\rangle)$ where $n \geq k$. The code is said to have distance d if the original state can be recovered from the encoded state that has transformed under any quantum operation which acts on at most $(d-1)/2$ qubits. Given an $[[m,1]]$ QECC with map Enc, we abuse notation and also write Enc for the corresponding $[[mn,n]]$ encoding that is obtained by applying Enc to each of the qubits in an n-qubit system.

We use $\underline{A}_n^k$ to denote the set of k distinct numbers between 1 and n through this section. Then $\underline{A}^k := \bigcup_{n \geq k} \underline{A}_n^k$ is the set of k distinct numbers. Given a k-qubit logical gate U and an element $\underline{a} = (a_1, \ldots, a_k) \in \underline{A}^k$, let $U(\underline{a})$ denote the gate U applied to qubits $a_1, \cdots a_k$.

Below we recall the definition of simulatable codes introduced in [23].

▶ **Definition 14.** *Given a k-qubit logical gate U and a quantum error-correcting code $\mathcal{C} = [[m,1]]$, let (σ_U, σ'_U) be a pair of states, and let ℓ be a positive integer. For each $1 \leq i \leq \ell$, let $\mathcal{O}_i$ be a mapping from elements $\underline{a} = (a_1, \cdots, a_k)$ in $\underline{A}^k$ to unitaries $\mathcal{O}_i(\underline{a})$ acting only on*

(i) *the physical qubits of codewords in $\mathcal{C}$ that corresponds to logical qubits $a_1, \cdots, a_k$, and*
(ii) *the register that holds σ_U.*

We say the tuple $(\sigma_U, \sigma'_U, \ell, \mathcal{O}_1, \cdots, \mathcal{O}_\ell)$ is an encoding of U in code $\mathcal{C}$ if

$$(\mathcal{O}_\ell(\underline{a})\dots\mathcal{O}_1(\underline{a}))(Enc(\rho)\otimes\sigma_U)(\mathcal{O}_\ell(\underline{a})\dots\mathcal{O}_1(\underline{a}))^* = Enc\big(U(\underline{a})\rho U(\underline{a})^*\big)\otimes\sigma'_U \tag{2}$$

for all $n \geq k$, elements $\underline{a} \in \underline{A}_n^k$, and n-qubit states ρ. If in addition, the unitaries $\mathcal{O}_1(\underline{a}), \cdots, \mathcal{O}_\ell(\underline{a})$ are gates in some set $\mathcal{U}$ for all $\underline{A} \in \underline{A}^k$, then we say the encoding $(\sigma_U, \sigma'_U, l, \mathcal{O}_1, \cdots, \mathcal{O}_\ell)$ uses physical gates in $\mathcal{U}$.

Given a circuit of logical gates $V = U_1 \dots U_k$ we refer to an encoding of V as the corresponding circuit of physical gates obtained by applying an encoding of each gate U_i.

▶ **Definition 15.** *An encoding $(\sigma_U, \sigma'_U, \mathcal{O}_1, \dots, \mathcal{O}_\ell)$ of a k-qubit logical gate U in a QECC $\mathcal{C}$ is called s-simulatable if for all $0 \leq t \leq \ell$, n-qubit states ρ, and subsets S of the physical qubits of $Enc(\rho)\otimes\sigma_U$ with $|S| \leq s$, the partial trace*

$$\mathrm{Tr}_{\overline{S}}\Big(\mathcal{O}_t(\underline{a})\dots\mathcal{O}_1(\underline{a}))(Enc(\rho)\otimes\sigma_U)(\mathcal{O}_t(\underline{a})\dots\mathcal{O}_1(\underline{a}))^*\Big)$$

is a $2^{|S|} \times 2^{|S|}$ matrix whose entries are rational and can be computed in polynomial time from t, $\underline{a}$ and S. In particular, this matrix is independent of ρ if $\mathcal{C}$ can correct arbitrary errors on s qubits.

▶ **Theorem 16** (Theorem 6 in [23]). *Let $\mathcal{U} = \{H, \Lambda(X), \Lambda^2(X)\}$. For every $s \in \mathbb{N}$, there exists a constant $n \in \mathbb{N}$ and a $[[n, 1]]$ QECC $\mathcal{C}$ such that any logical gate in $\mathcal{U}$ has an s-simulatable encoding in $\mathcal{C}$ using physical gates in $\mathcal{U}$.*

3 Low-weight Pauli braiding test and its rigidity

For any $a \in \{0,1\}^n$ and $W \in \{X, Z\}^n$, we use $W(a)$ to denote the sequence $W_1^{a_1}W_2^{a_2}\cdots W_n^{a_n}$ where $X^0 = Z^0 = I$. Let $\mathcal{I}_A := \{W(a) : W \in \{X,Z\}^n, a \in \{0,1\}^n$ such that $|a| \leq 6\}$ and let $\mathcal{I}_B := \{(W(a), W(a')) : W \in \{X,Z\}^n, a, a' \in \{0,1\}^n$ such that $|a|, |a'| \leq 6\}$ be the question sets for Alice and Bob respectively. We first describe the low-weight linearity test in Figure 3.

1. The verifier selects uniformly at random $W \in \{X, Z\}^n$ and strings $a, a' \in \{0,1\}^n$ satisfying $|a|, |a'| \leq 6$ (*i.e.* a, a' both have at most 6 non-zero entries).
2. The verifier sends $(W(a), W(a'))$ to Bob. If $a + a'$ has weight at most 6 then the verifier selects $W' \in \{W(a), W(a'), W(a+a')\}$ uniformly at random to send to Alice. Otherwise, the verifier uniformly at random sends $W' \in \{W(a), W(a')\}$ to Alice.
3. The verifier receives two bits (b_1, b_2) from Bob and one bit c from Alice.
4. If Alice receives $W(a)$ then the verifier requires $b_1 = c$. If Alice receives $W(a')$ then the verifier requires $b_2 = c$. If Alice receives $W(a + a')$ then the verifier requires $b_1 + b_2 = c$.

Figure 3 Low-weight linearity test.

Next, we introduce a natural version of the anti-commutation test in Figure 4. This test is built from the well-known Magic Square game which we described in Section 2.2.

Combining the low-weight linearity test and low-weight anti-commutation test, we now construct the low-weight Pauli braiding test and state its rigidity result.

1. The verifier samples uniformly at random a string $a \in \{0,1\}^n$ with exactly two non-zero entries $i < j$. The verifier also samples a row or column $q \in \{r_1, r_2, r_3, c_1, c_2, c_3\}$, and a variable v_k contained in q as in the Magic Square game.
2. Bob receives the question (q, a).
3. If $k \neq 9$ then Alice receives $W(a) = I^{i-1} W_i I^{j-i} W_j I^{n-j} \in \mathcal{I}_A$ with $\sigma_{W_i} \otimes \sigma_{W_j} = A_k$. If $k = 9$ then Alice receives question (v_9, a).
4. The players win if and only if Bob responds with a satisfying assignment to q and Alice provides an assignment to variable v_k that is consistent with Bob's.

Figure 4 Low-weight anti-commutation test.

▶ **Definition 17.** *The low-weight Pauli braiding test (LWPBT) is played by executing with probability* $1/2$ *either the low-weight anti-commutation test or the low-weight linearity test.*

The n-qubit LWPBT has a canonical perfect strategy $\widetilde{S}$ in which Alice and Bob share the n-qubit EPR pair $|\Phi^{\otimes}_{EPR}\rangle$ and Alice perform $\sigma_W(a) := \otimes_{i=1}^n \sigma_{W_i}^{a_i}$ on question $W(a) \in \mathcal{I}_A$. We have the following rigidity result for near-perfect strategies of LWPBT.

▶ **Theorem 18.** *There exists a constant* $C_{\mathrm{lw}} > 0$ *such that the following holds. For any* $\varepsilon > 0$, $n \in \mathbb{N}$, *and strategy* $\mathcal{S} = (\tau^A, \tau^B, |\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B)$ *for the* n*-qubit LWPBT with winning probability* $1 - \varepsilon$, *there are isometries* $V_A : \mathcal{H}_A \to (\mathbb{C}^2)^{\otimes n} \otimes \mathcal{H}_A^{aux}$, $V_B : \mathcal{H}_B \to (\mathbb{C}^2)^{\otimes n} \otimes \mathcal{H}_B^{aux}$ *and a unit vector* $|aux\rangle \in \mathcal{H}_A^{aux} \otimes \mathcal{H}_B^{aux}$ *such that*

$$\|(V_A \otimes V_B)\left(\tau^A(W(a)) \otimes Id_{\mathcal{H}_B}|\psi\rangle\right) - \left(\sigma_W(a) \otimes Id_{\mathbb{C}^{2^n}}|\Phi_{EPR}^{\otimes n}\rangle\right) \otimes |aux\rangle\| \leq C_{\mathrm{lw}} n^6 \varepsilon^{1/4}$$

for all $W(a) \in \mathcal{I}_A$.

The proof Theorem 18 uses a group-theoretical approach and can be found in the full version of our paper [9]. The idea is that we can round an approximate homomorphism (defined by a near-perfect stategy) of the Weyl-Heisenberg group to an exact representation using an enhanced Gowers-Hatami theorem.

3.1 An enhanced Gowers-Hatami theorem

For a finite group G, we use $\mathrm{Irr}(G)$ to denote the unique (up to unitary equivalence of elements) complete set of inequivalent irreducible representations. Given a finite group G, a function $f : G \to \mathcal{U}(\mathcal{H})$ from $\mathcal{G}$ to unitaries on a Hilbert space $\mathcal{H}$, and an irreducible representation $\phi : G \to \mathcal{U}(\mathbb{C}^d)$, the *Fourier transform of* f *at* ϕ is the operator

$$\widehat{f}(\phi) := \frac{1}{|G|} \sum_{g \in G} f(g) \otimes \overline{\phi(g)}, \tag{3}$$

where $\overline{\phi(g)}$ is the conjugate of the matrix $\phi(g) \in M_d(\mathbb{C})$ in the standard basis.

Let $f : G \to \mathcal{U}(\mathcal{H})$ be a function of a finite group G. Given a quantum state ρ on $\mathcal{H}$ and a positive real number ε, we say f is an (ε, ρ)*-homomorphism* provided that $f(g^{-1}) = f(g)^*$ and $\frac{1}{|G|} \sum_{h \in G} \|f(g)f(h) - f(gh)\|_\rho^2 \leq \varepsilon$ for all $g \in G$. In this case, by the well-known Gowers-Hatami theorem [21, 46], there is a Hilbert space $\mathcal{K}$, an isometry $V : \mathcal{H} \to \mathcal{K}$, and a representation $\phi : G \to \mathcal{U}(\mathcal{K})$ such that $\|f(g) - V^* \phi(g) V\|_\rho \leq \varepsilon$ for all $g \in G$. The following enhanced version of this theorem allows us to disregard all one-dimensional irreducible representations of the Weyl-Heisenberg group. Earlier works dealt with these one-dimensional representations by invoking a truncation of the isometry given by the Gowers-Hatami theorem. Unfortunately, in general, truncation of an isometry can fail to be an isometry.

▶ **Theorem 19.** *For any (ε, ρ)-homomorphism $f : G \to \mathcal{U}(\mathcal{H})$ of a finite group G on a finite-dimensional space $\mathcal{H}$, there exists a finite-dimensional Hilbert space $\mathcal{K}$, an isometry $I : \mathcal{H} \to \mathcal{K}$, and a representation $\pi : G \to \mathcal{U}(\mathcal{K})$ such that*

(i) $\|f(g) - I^*\pi(g)I\|_\rho^2 \leq \varepsilon$ *for all* $g \in G$, *and*

(ii) $\xi \in \mathrm{Irr}(G)$ *is an irreducible constituent of* π *only if* $\widehat{f}(\xi) \neq 0$.

We refer the readers to the full paper [9] for the proof details.

4 Modified Hamiltonian game

In this section, we show that for some local Hamiltonian H, one can construct a nonlocal game $\mathcal{G}(H)$ whose winning probability is closely related to the ground state energy $\lambda_0(H)$ of H. Our game is based on the Hamiltonian game introduced by Grilo [22]. We employ LWPBT against dishonest quantum provers and then perform parallel repetition to achieve a constant completeness-soundness gap. To incorporate LWPBT in our modified Hamiltonian test, we consider Hamiltonians with specific structures:

▶ **Definition 20.** *We say a Hamiltonian H is of XZ-type if it can be decomposed as $H = \frac{1}{m}\sum_{\ell=1}^{m} \gamma_\ell H_\ell$ where each $\gamma_\ell \in [-1, 1]$ and each term H_ℓ is a tensor product of operators σ_X, σ_Z or σ_I.*

Next, we define the relevant energy test which is analogous to the energy test used in [22].

▶ **Definition 21** (Energy test). *Given an n-qubit 6-local Hamiltonian $H = \frac{1}{m}\sum_{\ell=1}^{m} \gamma_\ell H_\ell$ of XZ-type we define the following energy test:*

1. *The verifier picks a term H_ℓ for $\ell \in [m]$ taken uniformly at random, and selects uniformly at random from the pairs $\{(W, r) \in \{X, Z\}^n \times \{0, 1\}^n : \sigma_W(r) = H_\ell\}$.*
2. *The verifier sends $W(r)$ to Alice, and tells Bob that the players are playing the energy test.*
3. *Alice responds with a single value $c \in \{-1, 1\}$ and Bob responds with $2n$ bits $a_1, \ldots a_n, b_1, \ldots b_n$.*
4. *The verifier next computes bit string d as follows. Take $d_i = (-1)^{a_i}$ if $r_i = 1$ and $W_i = X$, take $d_i = (-1)^{b_i}$ if $r_i = 1$ and $W_i = Z$, and take $d_i = 0$ in all other cases.*
5. *The verifier accepts if $c \cdot \prod_i d_i \neq sign(\gamma_l)$, and rejects with probability $|\gamma_l|$ otherwise.*

Combining the LWPBT and Energy test we define our modified Hamiltonian test:

▶ **Definition 22** (Hamiltonian test). *Let $H = \frac{1}{m}\sum_{\ell=1}^{m} \gamma_\ell H_\ell$ be a k-local Hamiltonian of XZ-type and let $p \in (0, 1)$. We define the following game $\mathcal{G}(H, p)$: with probability $(1 - p)$ the players play LWPBT introduced in Section Section 3, and with probability p the players play energy test described in Definition 21.*

We refer to a strategy $\mathcal{S}$ for $\mathcal{G}(H, p)$ as a *semi-honest strategy* if the players employ the canonical perfect strategy when playing LWPBT. Hence in a semi-honest strategy Alice and Bob hold n EPR pairs and Alice must perform $\sigma_W(r)$ on question $W(r)$ since she cannot distinguish questions from LWPBT or energy test. We also define the *honest strategy* $\mathcal{S}_h$ for $\mathcal{G}(H, p)$ in which the players employ the canonical perfect strategy when playing LWPBT, and in the energy test, Bob honestly teleports the ground state of H to Alice and provides the verifier with the teleportation keys.

Below we analyze the players' ability to win the overall game $\mathcal{G}(H,p)$ assuming the players are using a semi-honest strategy.

▶ **Lemma 23** (Lower bound on semi-honest strategies). *Suppose $H = \frac{1}{m}\sum_{l=1}^{m}\gamma_l H_l$ is an n-qubit 6-local XZ Hamiltonian, and Alice and Bob are performing a semi-honest strategy $\mathcal{S}$ for $\mathcal{G}(H,p)$. Then*

$$\omega(\mathcal{S}) \le \omega(\mathcal{S}_h) = 1 - p(\frac{1}{2m}\sum_{l}|\gamma_l| + \frac{1}{2}\lambda_0(H)).$$

Proof. Suppose the players are employing a semi-honest strategy $\mathcal{S} = (\tau^A, \tau^B, |\psi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B)$ for $\mathcal{G}(H,p)$. Let $\tau := \mathrm{Tr}_{\mathcal{H}_B}(|\psi\rangle\langle\psi|)$. Since the players win the LWPBT perfectly, they can only lose the overall game if they are playing an instance of the energy test. Let $a, b \in \{0,1\}^n$ be the answers Bob provides in the energy test, and let $\rho := \sigma_X^b\sigma_Z^a\tau\sigma_Z^a\sigma_X^b$.

Suppose in a round of the energy test, the verifier picks an $\ell \in [m]$ and selects a $W(r)$ for Alice. As discussed above, since Alice cannot distinguish questions from LWPBT and the energy test, she must perform $\sigma_W(r) = H_\ell$ on her registers. Hence $\mathbb{E}(c \cdot \prod_i d_i) = \mathrm{Tr}(H_\ell\sigma_X^b\sigma_Z^a\tau\sigma_Z^a\sigma_X^b) = \mathrm{Tr}(H_\ell\rho)$. Let p_ℓ be the probability of $c\prod_i d_i = sign(\gamma_\ell)$. Then $\mathbb{E}(c\prod_i d_i) = p_\ell sign(\gamma_\ell) - (1-p_\ell)sign(\gamma_\ell)$, or in other words, $\gamma_\ell\mathbb{E}(c\prod_i d_i) = (2p_\ell - 1)|\gamma_\ell|$. This implies the verifier rejects with probability

$$p_\ell|\gamma_\ell| = \frac{|\gamma_\ell| + \gamma_\ell\mathbb{E}(c\prod_i d_i)}{2} = \frac{|\gamma_\ell| + \gamma_\ell\mathrm{Tr}(H_\ell\rho)}{2}.$$

Thus by averaging over $\ell \in [m]$ we see that the players lose the energy test with probability

$$\frac{1}{m}\sum_{\ell\in[m]}\frac{|\gamma_\ell| + \gamma_\ell\mathrm{Tr}(H_\ell\rho)}{2} = \frac{1}{2m}\sum_{\ell\in[m]}|\gamma_\ell| + \frac{1}{2}\mathrm{Tr}(H\rho).$$

This probability is minimized if and only if ρ is indeed the density matrix of the ground state of H and in such case, the probability of winning the overall game is at most

$$1 - p\big(\frac{1}{2m}\sum_{\ell\in[m]}|\gamma_\ell| + \frac{1}{2}\lambda_0(H)\big).$$

This probability can be achieved if Bob teleports over the ground state and supplies the verifier with the verification keys in the energy test. ◀

▶ **Lemma 24** (Upper bound on dishonest strategies). *Let $H = \frac{1}{m}\sum_{\ell=1}^{m}\gamma_\ell H_\ell$ be a 6-local, n-qubit Hamiltonian of XZ-type. For any $\eta \in (0,1)$, let $p = \frac{4n^{-6}\eta^{3/4}}{(C_{\mathrm{lw}}+1)3^{3/4}}$ where C_{lw} is the constant given in Theorem 18. Then $\omega^*\big(\mathcal{G}(H,p)\big) \le \omega(\mathcal{S}_h) + \eta$, where $\omega(\mathcal{S}_h) = 1 - p\big(\frac{1}{2m}\sum_{\ell\in[m]}|\gamma_\ell| + \frac{1}{2}\lambda_0(H)\big)$ as in Lemma 24.*

Proof. Suppose the provers are employing a strategy $\mathcal{S} = (\tau^A, \tau^B, |\psi\rangle)$ for $\mathcal{G}(H,p)$ that wins LWPBT with probability $1-\varepsilon$ and wins the energy test with probability $\delta + 1 - \frac{\sum_\ell|\gamma_\ell|}{2m} - \frac{\lambda_0(H)}{2}$. Theorem 18 implies $\delta \le C_{\mathrm{lw}}n^6\varepsilon^{1/4}$. Then for $p := \frac{4n^{-6}\eta^{3/4}}{(C_{\mathrm{lw}}+1)3^{3/4}}$ with $\eta \in (0,1)$, we have $\eta + \varepsilon = \eta/3 + \eta/3 + \eta/3 + \varepsilon \ge 4(\frac{\eta^3\varepsilon}{3^3})^{1/4} = p(C_{\mathrm{lw}}+1)n^6\varepsilon^{1/4}$. It follows that

$$p\delta - (1-p)\varepsilon \le pCn^6\varepsilon^{1/4} + p\varepsilon - \varepsilon \le pC_{\mathrm{lw}}n^6\varepsilon^{1/4} + pn^6\varepsilon^{1/4} - \varepsilon = p(C_{\mathrm{lw}}+1)n^6\varepsilon^{1/4} - \varepsilon \le \eta.$$

Hence the overall winning probability is given by

$$\omega(\mathcal{S}) = (1-p)(1-\varepsilon) + p(\delta + 1 - \frac{\sum_\ell|\gamma_\ell|}{2m} - \frac{\lambda_0(H)}{2}) = \omega(\mathcal{S}_h) + p\delta - (1-p)\varepsilon \le \omega(\mathcal{S}_h) + \eta.$$

This completes the proof. ◀

In the rest of this paper, given an n-qubit, 6-local Hamiltonian H of XZ-type and parameters α and β with $\beta - \alpha \geq 1/poly(n)$, we use $\mathcal{G}(H)$ to denote the game $\mathcal{G}(H, p)$ with $p = \frac{32n^{-6}(\beta-\alpha)^{24}}{27(C_{\mathrm{lw}}+1)^4}$.

▶ **Theorem 25.** *Given an n-qubit, 6-local Hamiltonian $H = \frac{1}{m}\sum_{\ell=1}^{m}\gamma_\ell H_\ell$ and parameters α, β with $\beta - \alpha \geq 1/poly(n)$, let ω_α (resp. ω_β) denote the maximum winning probability for $\mathcal{G}(H)$ when $\lambda_0(H) \leq \alpha$ (resp. $\lambda_0(H) \geq \beta$). Then $\omega_\alpha - \omega_\beta \geq 1/poly(n)$.*

Proof. Let $\eta := \frac{16(\beta-\alpha)^{32}}{27(C_{\mathrm{lw}}+1)^4}$, and let $p := \frac{4n^{-6}\eta^{3/4}}{(C_{\mathrm{lw}}+1)3^{3/4}}$. Then $p = \frac{32n^{-6}(\beta-\alpha)^{24}}{27(C_{\mathrm{lw}}+1)^4}$, and hence $\mathcal{G}(H) = \mathcal{G}(H, p)$. By Lemma 23 and Lemma 24 we have $\omega(\mathcal{S}_h) \leq \omega^*\big(\mathcal{G}(H,p)\big) \leq \omega(\mathcal{S}_h) + \eta$. This implies $\omega_\alpha \geq 1 - p(\frac{1}{2m}\sum_\ell|\gamma_\ell| + \frac{1}{2}\alpha)$ and $\omega_\beta \leq 1 - p(\frac{1}{2m}\sum_\ell|\gamma_\ell| + \frac{1}{2}\beta) + \eta$. Since $n^6(\beta-\alpha)^7 \leq O(n^{-1})$, it follows that

$$\omega_\alpha - \omega_\beta \geq \frac{1}{2}p(\beta-\alpha) - \eta = \frac{16n^{-6}(\beta-\alpha)^{25}}{27(C_{\mathrm{lw}}+1)^4}(1 - n^6(\beta-\alpha)^7) \geq \frac{16n^{-6}(\beta-\alpha)^{25}}{27(C_{\mathrm{lw}}+1)^4}.$$

Hence $\omega_\alpha - \omega_\beta \geq 1/poly(n)$. ◀

We apply a threshold parallel repetition theorem due to Yuen for the gap amplification. For every $n \in \mathbb{N}$, let $\mathcal{G}(H), w_\alpha$ and w_β be as in Theorem 25. By [51, Theorem 41], there exists a $poly(n)$-computable transformation, called *anchoring*, that transforms $\mathcal{G}(H)$ to a two-player game $\mathcal{G}(H)_\perp$ with winning probability $1 - \frac{1-w^*(\mathcal{G}(H))}{2}$. So $w^*(\mathcal{G}(H)_\perp) = \begin{cases} 1-\varepsilon_\alpha/2 & \text{if } \lambda_0(H) \leq \alpha \\ 1-\varepsilon_\beta/2 & \text{if } \lambda_0(H) \geq \beta \end{cases}$, where $\varepsilon_\alpha := 1 - w_\alpha$ and $\varepsilon_\beta := 1 - w_\beta$. Then by [51, Theorem 42], there is a universal constant $C > 0$ such that for all integer $m \geq 1$, and $\gamma \geq 0$, the probability that in the game $\mathcal{G}(H)_\perp^m$ the players can win more than $\big(w^*(\mathcal{G}(H)_\perp) + \gamma\big)m$ games is at most $(1 - \gamma^9/2)^{Cm}$. Take $\gamma := \frac{\varepsilon_\alpha - \varepsilon_\beta}{4}$ and $m = \max\{4\gamma^{-2}, 2C\gamma^{-9}\}$. Let $\widehat{\mathcal{G}}(H) := \mathcal{G}(H)_\perp^m$ be the m parallel repeated anchoring version of $\mathcal{G}(H)$. We show that this nonlocal game has a constant completeness-soundness gap.

▶ **Theorem 26.** *Let $\widehat{\omega}_\alpha$ (resp. $\widehat{\omega}_\beta$) be the maximum winning probability for $\widehat{\mathcal{G}}(H)$ when $\lambda_0(H) \leq \alpha$ (resp. $\lambda_0(H) \geq \beta$). Then $\widehat{\omega}_\alpha - \widehat{\omega}_\beta \geq 1/4$.*

Proof. If $\lambda_0(H) \geq \beta$, then $\widehat{\omega}_\beta \leq (1 - \frac{\gamma^9}{2})^{Cm} \leq (1 - \frac{\gamma^9}{2})^{2/\gamma^9} < e^{-1} < 1/2$. Now suppose $\lambda_0(H) \leq \alpha$. An optimal strategy S for $\mathcal{G}(H)_\perp$ has winning probability $1 - \frac{\varepsilon_\alpha}{2}$. Let X be the random variable for the number of games the strategy S^m wins. Then $X \sim \mathrm{Binomial}(m, 1 - \frac{\varepsilon_\alpha}{2})$, so $\mathbb{E}X = m(1 - \frac{\varepsilon_\alpha}{2})$ and $\mathrm{Var}X = m\frac{\varepsilon_\alpha}{2}(1 - \frac{\varepsilon_\alpha}{2})$. Since $(1 - \frac{\varepsilon_\alpha}{2}) - (1 - \frac{\varepsilon_\beta}{2} + \gamma) = \frac{\varepsilon_\beta - \varepsilon_\alpha}{2} - \gamma = \gamma$, we obtain that

$$Pr(X \leq (1 - \tfrac{\varepsilon_\beta}{2} + \gamma)m) \leq Pr(|X - \mathbb{E}X| \geq \gamma m) \leq \frac{m(1 - \frac{\varepsilon_\alpha}{2})\frac{\varepsilon_\alpha}{2}}{(\gamma m)^2} = \frac{1}{m\gamma^2} \leq 1/4.$$

This implies $\widehat{\omega}_\alpha \geq w(S^m) = 1 - Pr(X \leq (1 - \frac{\varepsilon_\beta}{2} + \gamma)m) \geq 3/4$, so the theorem follows. ◀

5 Zero-knowledge proof system

In this section, we show that the family of games described in Definition 22 provides a statistical zero-knowledge MIP$^*[2,1]$ protocol for QMA with inverse polynomial completeness/soundness gap.

5.1 Simulation of history states for XZ-Hamiltonians

Before we introduce our MIP* protocol and proceed to our result on zero-knowledge, we reformulate a result, originally introduced by Broadbent and Grilo [5] (Lemma 3.5), so that it is more amenable to device-independent techniques.

▶ **Theorem 27** (Simulation of history states). *For any language $L = (L_{yes}, L_{no})$ in* QMA *and $s \in \mathbb{N}$, there is a family of verification circuits $V_x^{(s)} = U_T \dots U_1$ for L that acts on a witness of size $p(|x|)$ and on $q(|x|)$ ancillary qubits such that there exists a polynomial-time deterministic algorithm $Sim_{V^{(s)}}$ that takes as input an instance $x \in L$ and a subset $S \subseteq [T + p + q]$ with $|S| \leq 3s + 2$, then outputs a classical description of an $|S|$-qubit density matrix $\rho(x, S)$ with the following properties:*

1. *If $x \in L_{yes}$, then there exists a $p(|x|)$-qubit witness ψ^s such that $V_x^{(s)}$ accept with probability at least $1 - negl(n)$ on ψ^s and $\|\rho(x, S) - \mathrm{Tr}_{\overline{S}}(\rho)\|_{tr} \leq negl(|x|)$, where*
$$\rho = \frac{1}{T+1} \sum_{t,t' \in [T+1]} |unary(t)\rangle \langle unary(t)| \otimes U_t \dots U_1(\psi^s \otimes |0\rangle \langle 0|^{\otimes q})U_1^* \dots U_{t'}^*$$
is the history state of $V_x^{(s)}$ on witness ψ^s.
2. *Let H_i be one of the terms from the circuit-to-local Hamiltonian construction from $V_x^{(s)}$, and let S_i be the set of qubits on which H_i acts non-trivially. Then $\mathrm{Tr}(H_i \rho(x, S_i)) = 0$ for all $x \in L$.*
3. *The Hamiltonian H from the circuit-to-local Hamiltonian construction is a 6-local Hamiltonian of XZ type.*

The first two points were proven by Broadbent and Grilo using simulatable codes constructed from a different set of physical gates [5]. The last point follows from a similar approach in [25, Lemma 22]. A detailed proof can be found in the full version of our paper [9].

Below we only need to invoke Theorem 27 for the case of $s = 2$ in order to our zero-knowledge protocol. We use V_x to denote $V_x^{(2)}$ throughout the rest of this section.

5.2 A two prover zero-knowledge proof system for QMA

Let $L = (L_{yes}, L_{no})$ be a language in QMA. Figure 5 describes a two-prover one-round interactive proof system for L with a constant polynomial completeness-soundness gap.

$$x \xrightarrow{\text{Theorem 27}} V_x \xrightarrow{\text{circuit-to-Hamiltonian}} H_x \xrightarrow{\text{Definition 22}} \mathcal{G}(H_x) \xrightarrow{\text{Theorem 26}} \widehat{\mathcal{G}}_x := \widehat{\mathcal{G}}(H_x)$$

Figure 5 x is an instance in $L \in$ QMA. V_x is a $poly(|x|)$-size quantum circuit. H_x is a $poly(|x|)$-qubit 6-local Hamiltonian of XZ-type. $\widehat{\mathcal{G}}_x$ is a nonlocal game with $poly(|x|)$-bit questions and $poly(|x|)$-bit answers.

To prove the above interactive proof system for L has the statistical zero-knowledge property, we first establish that any malicious verifier $\widehat{V}$ and $x \in L_{yes}$, there exists a PPT simulator that can sample from $View(\widehat{V}(x), \mathcal{S}_h)$, where $\mathcal{S}_h$ is the honest strategy for $\mathcal{G}_x$ defined in Section 4.

▶ **Lemma 28.** *Suppose $x \in L_{yes}$ for some language $L = (L_{yes}, L_{no})$ in* QMA*. Let $\mathcal{G}_x$ be the corresponding nonlocal game described in Figure 5, and let $\mathcal{S}_h$ be the honest strategy for $\mathcal{G}_x$ defined in Section 4. For any malicious verifier $\widehat{V}$ there exists a PPT algorithm $Sim_{\widehat{V}}$ with output distribution negligibly close to $View(\widehat{V}(x), \mathcal{S}_h)$,* The proof of this lemma can be found in the full version of our paper [9]. All that remains is to argue that the interactive protocol described in Figure 5 based on the scaled-up game $\widehat{\mathcal{G}}_x$ is statistically zero-knowledge.

▶ **Theorem 29.** *The protocol described in Figure 5 is statistical zero-knowledge and has a constant completeness-soundness gap.*

Proof. The constant completeness-soundness gap follows directly from Theorem 26. To show the statistical zero-knowledge, we first consider the anchoring procedure for the game $\mathcal{G}_x$. We can specify an honest strategy $\mathcal{S}_{h,\perp}$ for the anchored version of $\mathcal{G}_x$ by fixing a choice of output for either player who receives question $\perp$ in the honest strategy. Then, given any malicious verifier $\widehat{V}(x)$, the simulator given in Theorem 29 can be trivially modified to sample from a distribution which is negligibly close to $View(\widehat{V}(x), \mathcal{S}_{h,\perp})$.

In the case of the threshold parallel repeated game $\widehat{\mathcal{G}}_x$, the honest strategy $\mathcal{S}^m_{h,\perp}$ is taken to be the m-fold product of the honest strategy $\mathcal{S}_{h,\perp}$. Then, as commented in [23], since the protocol only queries each player once, a new simulator can be obtained by sampling according to the m-fold product of the simulator used in the above lemma. ◀

6 Off-the-shelf model

6.1 Formal description of the model

Here we provide a formal description of the OTS model. This model is defined as a refinement of MIP*, where the completeness condition is weakened, allowing only one of the provers to be "all-powerful", while the other has limited functionality determined independently of the problem instance.

Off-the-shelf device. We first formalize the definition of a family of off-the-shelf devices. A *verification device* $D = (|\psi\rangle, \{P^1_a\}_a, \ldots, \{P^q_a\}_a)$ consists of a state $|\psi\rangle$ on Hilbert spaces $\mathcal{K}_A \otimes \mathcal{K}_B$ and a collection of POVMs $\{P^1_a\}_a, \ldots, \{P^q_a\}_a$ on $\mathcal{K}_A$. We say that a quantum strategy $\mathcal{S} = (\{E^x_a\}, \{F^y_b\}, |\phi\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B)$ can be *implemented* using D, if three conditions hold: (i) $\mathcal{H}_A = \mathcal{K}_A$ and $\mathcal{H}_B = \mathcal{K}_B \otimes \mathcal{K}_{B'}$ for some Hilbert space $\mathcal{K}_{B'}$. (ii) The set of measurements in $\mathcal{S}$ which act on $\mathcal{H}_A$ are a contained in D. (iii) The shared state $|\phi\rangle$ in $\mathcal{S}$ can be decomposed as $|\phi\rangle = |\psi\rangle \otimes |\phi'\rangle$ where $|\psi\rangle$ is the state in D and $|\phi'\rangle$ is some auxiliary state held on a Hilbert space $\mathcal{K}_{B'}$.

Given a collection of verification devices $\{D_n\}_{n\in\mathbb{N}}$, where each D_n consists of a state $|\psi_n\rangle$ and a sequence of POVMs, we say $\{D_n\}_{n\in\mathbb{N}}$ is an *efficient family of off-the-shelf devices* if there exists a polynomial-time uniform family of quantum circuits $\{Q_n\}_{n\in\mathbb{N}}$ satisfying the following: Q_n generates the state $|\psi_n\rangle$ from an all 0 state, and on input i measures $|\psi_n\rangle$ using the i-th POVM from D_n.

▶ **Definition 30.** *A promise language $L = (L_{yes}, L_{no})$ has an off-the-shelf (OTS) proof system if there exists an efficient family of off-the-shelf devices $\{D_n\}_{n\in\mathbb{N}}$, and a polynomial-time computable function that takes an instance x to the description of a non-local game $\mathcal{G}_x$ satisfying the following:*

1. ***Completeness using OTS devices.*** *For any $x \in L_{yes}$ with $|x| \leq n$, there exists a quantum strategy $\mathcal{S}_x$, which can be implemented using D_n, obtaining $\omega(\mathcal{G}_x, \mathcal{S}_x) \geq c$.*
2. ***Soundness.*** *For any $x \in L_{no}$ we have $\omega^*(\mathcal{G}_x) < s$.*

We use OTS *to denote the class of all languages L which admits an OTS proof system with a constant completeness-soundness gap.*

1. **Set-up:** The client sends a set-up parameter $k \in \mathbb{N}$ to the server who provides a verification device D_k from an efficient family of off-the-shelf devices $\{D_n\}_n$.
2. **Choice of computation:** The client sends a classical description of circuit Q, satisfying $|Q| \leq k$ to the server.
3. **Verifiable delegation:** The client plays a 1-round game $\widehat{\mathcal{G}}_Q$, using the server and device D_k as players. The client accepts if and only if the game is won.

Figure 6 A delegation protocol between a polynomial-time classical client and polynomial-time quantum server, who provides an untrusted verification device during set-up.

Any OTS proof system is described as a special instance of a 2-player, 1-round MIP^* proof system with additional constraints regarding the completeness condition and we say that an OTS proof system is statistically zero-knowledge if it is statistically zero-knowledge as an MIP^* proof system (see Definition 13 for details).

6.2 Applications to ZK and delegated computation

In this section, we show that any language in QMA admits a statistical zero-knowledge OTS proof system. We also consider how the OTS model can be scaled down to provide a protocol for verifiable delegated quantum computation.

▶ **Theorem 31.** *For every language L in* QMA*, there exists a statistical zero-knowledge OTS proof system for L with constant completeness and soundness gap.*

Proof. We will be working with the proof system sending an instance x to game $\widehat{\mathcal{G}}_x$, as described in Figure 5. Using the rigidity results in Section 3, we have already shown completeness and soundness of properties of the individual game $\widehat{\mathcal{G}}_x$ in Section 4. The ZK property of this game has also been shown in Section 5. All that remains to show is that this protocol further satisfies the extra restrictions of completeness using OTS devices outlined in Definition 30. That is, we need to show that there exists an efficient family of OTS devices $\{D_n\}$ which can implement the honest strategy $\mathcal{S}_x$ for all yes instances x.

For each $L \in \mathsf{QMA}$, there exists a polynomial f such that, for all $x \in L$ of size $|x| = n$ the corresponding Hamiltonian H_x is supported on at most $f(n)$ qubits. Next suppose $x \in L_{yes}$ with $|x| \leq n$. In the honest strategy for the game $\mathcal{G}_x$, Alice and Bob share at most $f(n)$-EPR pairs, additionally, Bob privately holds a ground state ρ for H_x. The measurements required by Alice always correspond to σ_X or σ_Z on up to 6 qubits of the shared EPR pairs, or $\sigma_X\sigma_Z \otimes \sigma_Z\sigma_X$ on two qubits. In the honest strategy $\mathcal{S}_x$ for the m-fold parallel repeated anchoring game $\widehat{\mathcal{G}}_x$, the players share $mf(n)$-EPR pairs and Alice's measures in σ_X or σ_Z on up to $6m$ qubits or measures with $\sigma_X\sigma_Z \otimes \sigma_Z\sigma_X$ on $2m$ qubits. Since $m = poly(n)$, we then satisfy the completeness condition required by specifying an efficient family of OTS devices $\{D_n\}$, where for each n the verification device D_n contains $mf(n)$-EPR pairs and all of the above required Pauli measurements on up to $6m$ qubits. ◀

In our application towards delegated quantum computation, we consider a novel type of interactive protocol, where in addition to exchanging classical messages, the server can send an untrusted verification device, as defined in Section 6.1, to the client (see Section 1). In Figure 6, we consider the case where in the first "message", called a set-up stage, the prover sends an untrusted verification device, which is followed by classical communication.

▶ **Theorem 32.** *For every language L in* BQP*, there is a statistical-zero-knowledge delegation protocol as outlined in Figure 6 for L with constant completeness and soundness gap.*

Proof (Sketch). We can view the BQP-complete problem from Definition 9 as a language in QMA. This allows us to apply the efficient mapping outlined in Figure 5 to obtain a corresponding game $\widehat{\mathcal{G}}_Q$. In this case, the ground state of the underlying Hamiltonian can be prepared by a polynomial-time quantum prover. Thus, as in the proof of Theorem 31, we can define the required polynomial-time uniform family of OTS devices $\{D_n\}_{n\in\mathbb{N}}$ by taking D_n to contain suitably many EPR pairs, as well as the required Pauli measurements. Since furthermore the required ground state can always be prepared by a polynomial-time quantum prover, an honest server can obtain the required completeness in Step 3 by generating this state and teleporting it to the verification device when required. We also have that the above delegation protocol inherits the ZK property via the results of Theorem 31. ◀

References

1 Dorit Aharonov, Michael Ben-Or, Elad Eban, and Urmila Mahadev. Interactive proofs for quantum computations, 2017. arXiv:1704.04487.

2 Michael Ben-Or, Oded Goldreich, Shafi Goldwasser, Johan Håstad, Joe Kilian, Silvio Micali, and Phillip Rogaway. Everything provable is provable in zero-knowledge. In *Advances in Cryptology – CRYPTO '88*, pages 37–56, 1988. doi:10.1007/0-387-34799-2_4.

3 Michael Ben-Or, Shafi Goldwasser, Joe Kilian, and Avi Widgerson. Multi-prover interactive proofs: how to remove intractability assumptions. In *STOC '88: Proceedings of the twentieth ACM symposium on Theory of computing*, pages 113–131, 1988. doi:10.1145/62212.62223.

4 Anne Broadbent. How to verify a quantum computation. *Theory of Computing*, 14(1):1–37, 2018. doi:10.4086/toc.2018.v014a011.

5 Anne Broadbent and Alex B. Grilo. QMA-hardness of consistency of local density matrices with applications to quantum zero-knowledge. *SIAM Journal on Computing*, 51(4):1400–1450, 2022. doi:10.1137/21M140729X.

6 Anne Broadbent, Gus Gutoski, and Douglas Stebila. Quantum one-time programs. In *Advances in Cryptology – CRYPTO 2013*, volume 2, pages 344–360, 2013. doi:10.1007/978-3-642-40084-1_20.

7 Anne Broadbent, Zhengfeng Ji, Fang Song, and John Watrous. Zero-knowledge proof systems for QMA. In *2016 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2016)*, pages 31–40, 2016. doi:10.1109/FOCS.2016.13.

8 Anne Broadbent, Zhengfeng Ji, Fang Song, and John Watrous. Zero-knowledge proof systems for QMA. *SIAM Journal on Computing*, 49(2):245–283, 2020. doi:10.1137/18M1193530.

9 Anne Broadbent, Arthur Mehta, and Yuming Zhao. Quantum delegation with an off-the-shelf device, 2023. arXiv:2304.03448.

10 André Chailloux and Anthony Leverrier. Relativistic (or 2-prover 1-round) zero-knowledge protocol for NP secure against quantum adversaries. In *Advances in Cryptology – EUROCRYPT 2017*, volume 3, pages 369–396, 2017. doi:10.1007/978-3-319-56617-7_13.

11 Rui Chao, Ben W. Reichardt, Chris Sutherland, and Thomas Vidick. Test for a large amount of entanglement, using few measurements. *Quantum*, 2:92, 2018. doi:10.22331/q-2018-09-03-92.

12 Alessandro Chiesa, Michael Forbes, Tom Gur, and Nicholas Spooner. Spatial isolation implies zero knowledge even in a quantum world. In *2018 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2018)*, pages 755–765, 2018. doi:10.1109/FOCS.2018.00077.

13 John F. Clauser, Michael A. Horne., Abner Shimony, and Richard A. Holt. Proposed experiment to test local hidden-variable theories. *Physical Review Letters*, 23(15):880–884, 1969. doi:10.1103/PhysRevLett.23.880.

14 Richard Cleve, Peter Høyer, Benjamin Toner, and John Watrous. Consequences and limits of nonlocal strategies. In *19th Annual Conference on Computational Complexity (CCC 2004)*, pages 236–249, 2004. doi:10.1109/CCC.2004.1313847.

15 Andrea Coladangelo. Parallel self-testing of (tilted) EPR pairs via copies of (tilted) CHSH and the magic square game. *Quantum Information & Computation*, 17(9&10):831–865, 2017. doi:10.26421/QIC17.9-10-6.

16 Andrea Coladangelo, Alex B Grilo, Stacey Jeffery, and Thomas Vidick. Verifier-on-a-leash: New schemes for verifiable delegated quantum computation, with quasilinear resources. In *Advances in Cryptology — EUROCRYPT 2019*, volume 3, pages 247–277, 2019. doi:10.1007/978-3-030-17659-4_9.

17 Claude Crépeau and John Stuart. Zero-knowledge MIPs using homomorphic commitment schemes, 2023. arXiv:2304.09784.

18 Joseph F. Fitzsimons and Elham Kashefi. Unconditionally verifiable blind quantum computation. *Physical Review A*, 96(1):012303, 2017. doi:10.1103/PhysRevA.96.012303.

19 Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(3):690–728, 1991. doi:10.1145/116825.116852.

20 Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989. doi:10.1137/0218012.

21 W. T. Gowers and O. Hatami. Inverse and stability theorems for approximate representations of finite groups. *Sbornik: Mathematics*, 208(12):1784–1817, 2017. doi:10.1070/sm8872.

22 Alex B. Grilo. A simple protocol for verifiable delegation of quantum computation in one round. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, pages 28:1–28:13, 2019. doi:10.4230/LIPIcs.ICALP.2019.28.

23 Alex B. Grilo, William Slofstra, and Henry Yuen. Perfect zero knowledge for quantum multiprover interactive proofs. In *2019 60th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2019)*, pages 611–635, 2019. doi:10.1109/FOCS.2019.00044.

24 Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. *Journal of the ACM*, 58(6):30, 2011. doi:10.1145/2049697.2049704.

25 Zhengfeng Ji. Classical verification of quantum proofs. In *STOC 2016: Proceedings of the 48th ACM SIGACT symposium on Theory of Computing*, pages 885–898, 2016. doi:10.1145/2897518.2897634.

26 Zhengfeng Ji. Compression of quantum multi-prover interactive proofs. In *STOC 2017: Proceedings of the 49th ACM SIGACT symposium on Theory of Computing*, pages 289–302, 2017. doi:10.1145/3055399.3055441.

27 Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. MIP*=RE, 2020. arXiv:2001.04383.

28 Joe Kilian. Founding cryptography on oblivious transfer. In *STOC '88: Proceedings of the twentieth ACM symposium on Theory of computing*, pages 20–31, 1988. doi:10.1145/62212.62215.

29 Alexei Yu. Kitaev, Alexander Shen, and Mikhail N. Vyalyi. *Classical and Quantum Computation.* American Mathematical Society, 2002. doi:10.1090/gsm/047.

30 Hirotada Kobayashi and Keiji Matsumoto. Quantum multi-prover interactive proof systems with limited prior entanglement. *Journal of Computer and System Sciences*, 66(3):429–450, 2003. doi:10.1016/S0022-0000(03)00035-7.

31 Urmila Mahadev. Classical verification of quantum computations. In *2018 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2018)*, pages 259–267, 2018. doi:10.1109/FOCS.2018.00033.

32 Laura Mančinska, Jitendra Prakash, and Christopher Schafhauser. Constant-sized robust self-tests for states and measurements of unbounded dimension, 2021. arXiv:2103.01729.

33 Dominic Mayers and Andrew Yao. Self testing quantum apparatus. *Quantum Information & Computation*, 4(4):273–286, 2004. doi:10.26421/QIC4.4-4.

34 M. McKague, T. H. Yang, and V. Scarani. Robust self-testing of the singlet. *Journal of Physics A*, 45(45):455304, 2012. doi:10.1088/1751-8113/45/45/455304.

35 Matthew McKague. Self-testing in parallel with CHSH. *Quantum*, 1:1, 2017. doi:10.22331/q-2017-04-25-1.

36 N. David Mermin. Simple unified form for the major no-hidden-variables theorems. *Physical Review Letters*, 65(27):3373–3376, 1990. doi:10.1103/PhysRevLett.65.3373.

37 Anand Natarajan and Thomas Vidick. A quantum linearity test for robustly verifying entanglement. In *STOC 2017: Proceedings of the 49th ACM SIGACT symposium on Theory of Computing*, pages 1003–1015, 2017. doi:10.1145/3055399.3055468.

38 Anand Natarajan and Thomas Vidick. Low-degree testing for quantum states, and a quantum entangled games PCP for QMA. In *2018 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2018)*, pages 731–742, 2018. doi:10.1109/FOCS.2018.00075.

39 Michael A. Nielsen and Issac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

40 Asher Peres. Incompatible results of quantum measurements. *Physics Letters A*, 151(3):107–108, 1990. doi:10.1016/0375-9601(90)90172-K.

41 Ben W. Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496:456–460, 2013. doi:10.1038/nature12035.

42 Gregory Rosenthal and Henry Yuen. Interactive proofs for synthesizing quantum states and unitaries. In *13th Conference on Innovations in Theoretical Computer Science—ITCS 2022*, pages 112:1–112:4, 2022. doi:10.4230/LIPIcs.ITCS.2022.112.

43 William Slofstra. The set of quantum correlations is not closed. *Forum of Mathematics, Pi*, 7:e1:1–e1:41, 2019. doi:10.1017/fmp.2018.3.

44 William Slofstra. Tsirelson's problem and an embedding theorem for groups arising from non-local games. *Journal of the American Mathematical Society*, 33:1–56, 2020. doi:10.1090/jams/929.

45 Boris S. Tsirelson. Some results and problems on quantum Bell-type inequalities. *Hadronic Journal Supplement*, 8:329–345, 1993.

46 Thomas Vidick. An expository note on "a quantum linearity test for robustly verifying entanglement", 2018. URL: http://users.cms.caltech.edu/~vidick/notes/pauli_braiding_1.pdf.

47 Thomas Vidick and Tina Zhang. Classical zero-knowledge arguments for quantum computations. *Quantum*, 4:266, 2020. doi:10.22331/q-2020-05-14-266.

48 John Watrous. Succinct quantum proofs for properties of finite groups. In *41st Annual Symposium on Foundations of Computer Science (FOCS 2000)*, pages 537–546, 2000. doi:10.1109/SFCS.2000.892141.

49 John Watrous. PSPACE has constant-round quantum interactive proof systems. *Theoretical Computer Science*, 292(3):575–588, 2003. doi:10.1016/S0304-3975(01)00375-9.

50 John Watrous. Quantum computational complexity. In *Encyclopedia of complexity and systems science*, pages 7174–7201. Springer, 2009. doi:10.1007/978-3-642-27737-5_428-3.

51 Henry Yuen. *Games, protocols, and quantum entanglement*. PhD thesis, Massachusetts Institute of Technology, 2016. URL: https://dspace.mit.edu/handle/1721.1/107364.